

Effizienter Local-Dimming Algorithmus und  
Matrix-Backlight-Designoptimierung für LC-Displays zur  
Steigerung der Energie-Effizienz und der Bildqualität mit  
Fokus auf automotive-spezifische Anforderungen

Dissertation  
zur Erlangung des Grades  
des Doktors der Ingenieurwissenschaften  
der Naturwissenschaftlich-Technischen Fakultät  
der Universität des Saarlandes

von  
Maxim Schmidt

Saarbrücken  
2022

Tag des Kolloquiums: 16.12.2022

Dekan: Univ.-Prof. Dr. rer. nat. Ludger Santen

Berichterstatter: Univ.-Prof. Dr.-Ing. Chihao Xu

Berichterstatter: Dr.-Ing. Andreas Karrenbauer

Berichterstatter: Univ.-Prof. Dr. rer. nat. Rolf Pelster

Vorsitz: Univ.-Prof. Dr. rer. nat. Alexandra K. Kiemer

Akad. Mitarbeiter: Dr.-Ing. Christian Bur

## Kurzzusammenfassung

Den wachsenden Anforderungen an Auto-Displays sollen mit LCDs mit hochauflösendem Backlight und einem adäquaten Local-Dimming (LD) Algorithmus begegnet werden. Als Basis werden der SSC LD-Algorithmus für Edge-Lit analysiert und seine Limitierungen, wie zeitliches Limit und steigende HW-Kosten bei Direct-Lit Anwendung, aufgezeigt. Ein Novum zur Lösung der Probleme ist die Entwicklung und der Einsatz von LSF-Modellen, um die Speichergrößen bei steigenden LED-Zahlen konstant zu halten. Zudem wird die Auswirkung der LSF-Form auf die LD-Performance untersucht.

Durch Einsatz des LD-Algorithmus im Backlight-Designprozess kann der Kompromiss zwischen der Energie-Effizienz und Anfälligkeit für Artefakte wie Inhomogenität oder Halo verbessert werden. Ein Design-Parameter wurde eingeführt, welcher den Kompromiss abbildet.

Neben der Energie-Effizienz ist die wahrgenommene Wertigkeit ein Schwerpunkt, die mit Features wie Boosting und 2D-Filter erhöht wird. Das Halo Artefakt wird abgeschwächt und der Ausfall einzelner LEDs wird weitgehend kompensiert. Die neuen Methoden werden in FPGAs auf der Pixelpipeline implementiert und mit Demonstratoren validiert. Eine hohe Verbesserung der Bildqualität, des Schwarzwertes sowie des Powersavings, welches für HMI Bilder je nach Steuerungsparametern zwischen 34 % und 54 % liegt, werden erreicht. Der neue Algorithmus in Kombination mit adäquatem BLU Design liefert ein USP für Auto-Displays welches als IP an Anwender übertragen werden kann.

## Abstract

The growing demands on automotive displays are to be met by LCDs with high-resolution backlight and an adequate local dimming (LD) algorithm. In this context, a high number of LEDs leads to challenges. As the starting point, the SSC LD algorithm for edge-lit is analyzed and its limitations such as time limit and increasing HW cost for direct lit displays are demonstrated. A novelty to solve the issues is the development of LSF models to keep the memory size constant while the LED numbers increase. In addition, the impact of LSF shape on LD performance is investigated.

By integrating the LD algorithm into the backlight design process, the tradeoff between energy efficiency and susceptibility to artifacts such as non-uniformity or halo can be lifted. A design parameter has been introduced that maps this trade-off.

In addition to energy efficiency, the focus is also on perceptual value of displays, which is increased with features such as boosting and 2D filters. Halo artifacts are mitigated, and single LED outage is largely compensated. The proposed methods are implemented on FPGAs within the pixel pipeline and validated with demonstrators. A high improvement of image quality, black level as well as powersaving is achieved. The last ranges from 34 % to 54 % for HMI images depending on the control parameters. The new algorithm in combination with an adequate BLU design provides a USP for automotive displays which can be transferred as an IP to users.



## Danksagung

An dieser Stelle möchte ich allen Personen danken, die mich wissentlich oder unwissentlich während meiner Studien- und Promotionszeit in verschiedenster Form unterstützt haben.

Mein größter Dank gilt meinem Doktorvater Univ.-Prof. Dr.-Ing. Chihao Xu für die Möglichkeit am Lehrstuhl für Mikroelektronik diese Arbeit im Bereich der Displayforschung anfertigen zu können. Im Laufe dieser Zeit habe ich nicht nur fachliche Kompetenzen erlangt, sondern mich auch persönlich weiterentwickeln und an den Herausforderungen wachsen können. Er stand mir immer mit Rat und neuen Denkansätzen zur Seite und ermöglichte die Teilnahme an vielen internationalen Fachkonferenzen.

Außerdem danke ich Dr.-Ing. Andreas Karrenbauer dem Leiter der Forschungsgruppe *Discrete Optimization* des *Max Planck Center for Visual Computing and Communication* und Senior Researcher der Abteilung *Algorithms and Complexity* am Max-Planck-Instituts für Informatik in Saarbrücken und einem der Erfinder des SSC-Algorithmus, für die Übernahme des Zweitgutachtens.

Ferner danke ich den weiteren Mitgliedern des Prüfungsausschusses.

Ich danke auch Ramazan Ayasli und Julian Ritter, sowie Daniel Schäfer, Xingtong Jiang, Michael Grüning, Pascal Volkert und Tobias Jung für die manchmal notwendigen Denkanstöße und die sehr lehrreichen „Fortbildungen“ in den Pausendiskussionen. Ebenfalls bedanke ich mich für die konstruktive Zusammenarbeit bei einigen Publikationen. Danke Daniel und Tobias für die vielen hilfreichen Tipps und Motivation zu Beginn meiner Arbeit am LME.

Auch danke ich Matthias Jost für die Unterstützung bei der Erstellung einiger *TikZ* Abbildungen in  $\LaTeX$  sowie den aktuellen und ehemaligen Hiwis und Studenten des Lehrstuhls. Silke Maas danke ich für die Übernahme so einiger Verwaltungsaufgaben und die nicht immer einfache Kommunikation mit anderen Abteilungen.

An dieser Stelle danke ich auch meinen Eltern Alla und Eugen für die persönliche und finanzielle Unterstützung während und nach meinem vorangegangenen Studium.

Mein besonderer Dank gilt meiner Frau Daniela. Sie hat mich von Beginn an stets unterstützt und motiviert und auch während der belastenden COVID-19-Pandemie mir den Rücken freigehalten. Auch unseren Kindern Katharina und Julia danke ich für den Verzicht auf die gemeinsame Zeit. Diese werden wir nachholen.



# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>1</b>
1.1. Motivation und Hinführung zum Thema . . . . .	2
1.2. Zielsetzung und Struktur der Arbeit . . . . .	5
<b>2. Grundlagen</b>	<b>11</b>
2.1. Radio- und Fotometrische Grundlagen . . . . .	11
2.2. Menschliche Wahrnehmung . . . . .	14
2.3. Aktuell verbreitete Displays und deren Funktion . . . . .	20
2.3.1. Aufbau und Funktion von LC-Displays . . . . .	21
2.3.2. Grundlegender Aufbau von OLED-Displays . . . . .	24
2.4. Imperfektion von LCDs . . . . .	26
2.5. Dimming Prinzipien . . . . .	29
2.5.1. Konzept . . . . .	29
2.5.2. Light-Spread-Function und Einflussmatrix . . . . .	32
2.5.3. Mathematische Beschreibung . . . . .	34
2.5.4. Pixelkompensation . . . . .	35
2.6. Hardwareumsetzung eines Bildverarbeitungsalgorithmus . . . . .	36
<b>3. Besonderheiten und Lösungsansätze in Automobil-Anwendungen</b>	<b>41</b>
3.1. Anforderungen an Displays im Fahrzeug . . . . .	41
3.2. OLED Displays in der Automobilanwendung . . . . .	45
3.3. Global-Dimming zur Reduktion des Strombedarfs . . . . .	47
3.4. Verbesserung des Global-Dimmings zur Qualitätssteigerung . . . . .	50
3.4.1. Konzept . . . . .	51
3.4.2. Globale Bildanalyse . . . . .	53
3.4.3. Hardwareimplementierung . . . . .	54
3.4.4. Ergebnisse . . . . .	57
3.5. Local-Dimming . . . . .	60
3.5.1. Edge-Lit Dimming (SSC) und Stand der Technik . . . . .	61
3.5.1.1. Preprocessing . . . . .	64
3.5.1.2. SSC-Optimierer mit Umsetzung der LSF und Pixelorder . . . . .	65
3.5.1.3. Postprocessing . . . . .	67
3.5.2. Direct-Lit Dimming und Stand der Technik . . . . .	69
3.5.3. Ansteuerschemata für eine hoch aufgelöste LED-Matrix . . . . .	72
3.6. Schlussfolgerungen . . . . .	74

<b>4. Hochauflösende Backlight-Units</b>	<b>75</b>
4.1. Untersuchung des Einflusses auf die Pixel und Einfluss auf Local-Dimming	77
4.1.1. Vergleich mit niedrig aufgelösten BLUs	78
4.1.2. Verschiedene LSF-Formen	86
4.2. Qualitätsmetriken für hochauflösende BLUs	92
4.2.1. Quantifizierung der Homogenität	94
4.2.2. Schwarze Flächen als Qualitätsmaß	97
4.3. LSF Modellierung für Simulations- und Bewertungszwecke	98
4.3.1. Gauß-Verteilung als LSF-Modell	101
4.3.2. Einführung eines neuen Modells für radial-invariante LSF-Formen	102
4.4. Designparameter einer BLU: Das FWHM/Pitch Verhältnis	105
4.5. Auswirkungen der LSF-Form auf die Robustheit und Qualität	109
4.5.1. Einfluss der LSF-Form bei konstantem FWHM/Pitch Verhältnis	111
4.5.2. Einfluss der LSF-Form mit konstantem maximalen Einfluss	117
4.6. Zusammenfassung	124
<b>5. Beschleunigte Optimierung, LSF-Modellkonzept und Systemarchitektur für eine effiziente Hardware-Implementierung des Local-Dimming Algorithmus</b>	<b>129</b>
5.1. Auswirkungen und Herausforderungen hoher LED-Auflösungen auf den Local-Dimming Algorithmus	129
5.1.1. Condenser- und Backlight Sample-Point Auflösung	130
5.1.2. Grenzen der Hardware-Implementierung und resultierende Herausforderungen	133
5.2. Konzept für eine effiziente Optimierung	135
5.2.1. Methode zur beschleunigten Optimierung	136
5.2.2. Betrachtung der Auswirkungen einer reduzierten LED-Auswahl	137
5.3. Entwicklung eines LUT-basierten LSF-Modellkonzepts zur Speicherreduktion	140
5.3.1. Idee und Konzept	141
5.3.2. Verfahren zur konsistenten Modellierung	147
5.3.3. Reduzierung auf Sample-Points	153
5.3.4. Verringerung des Modellspeichers durch reduzierte Stützstellendichte	156
5.4. Scanning Sequenz und Struktur für eine neue Pixelorder	158
5.4.1. Strukturelle und inhaltliche Erweiterung für den Hardware-Einsatz	159
5.4.2. Selektion der Scanning Sequenz	161
5.5. LED-Clustering	167
5.6. Neue Local-Dimming Systemarchitektur und Implementierung	169
5.6.1. Systemkonzept und Aufbau	171
5.6.2. Integration und Implementierung	174
5.6.2.1. Implementierung einer <i>Offline</i> -Prozessierung	174
5.6.2.2. <i>Echtzeit/Online</i> -Prozessierung	175
5.6.2.3. <i>LSF-Decoder</i> für <i>Optimizer</i> Core und Backlight Berechnung	177
5.6.2.4. Funktion weiterer Module im <i>Optimizer</i> und Integration des <i>LSF-Decoders</i>	185
5.6.2.5. Zeitlicher Ablauf und Übersicht	188
5.7. Ergebnisse	189

<b>6. Weitere Methoden zur Steigerung der Bildqualität</b>	<b>193</b>
6.1. Das Boosting Konzept . . . . .	194
6.1.1. Technische Umsetzung . . . . .	195
6.1.2. Resultate . . . . .	199
6.2. Entschärfung des Halo-Artefakts . . . . .	201
6.2.1. Beschreibung des Halos . . . . .	201
6.2.2. Unterdrückung von Halo . . . . .	203
6.2.2.1. Räumliche Filter für LEDs . . . . .	204
6.2.2.2. LED-Min Werte . . . . .	208
6.3. Algorithmenseitige Ansätze zur Verbesserung des Schwarzwertes in den Displayecken im Automobil . . . . .	211
6.3.1. Anpassung der Scanning Sequenz der Pixelorder . . . . .	211
6.3.2. Parameterwahl . . . . .	213
6.3.3. Corner-Dimming . . . . .	214
6.3.4. Ergebnisse der Parameterwahl und Corner-Dimming . . . . .	215
6.4. Erhaltung und Sicherstellung der visuellen Qualität bei LED-Ausfall . . . . .	217
6.5. Methoden zur Abdeckungen noch höherer BLU-Auflösungen . . . . .	219
6.5.1. Erhöhung der Systemfrequenz . . . . .	220
6.5.2. Betrachtung einer begrenzten LED-Zahl zur Sample-Point-Berechnung . . . . .	221
6.5.3. Resultate . . . . .	226
<b>7. Gesamte Betrachtung und Ausblick</b>	<b>229</b>
7.1. Betrachtung des Local-Dimming IPs . . . . .	229
7.2. Weitere Arbeiten und Ausblick . . . . .	235
<b>8. Zusammenfassung</b>	<b>239</b>
<b>A. Weitere Abbildungen</b>	<b>243</b>
<b>B. Übersicht Testbilder</b>	<b>247</b>
<b>C. Source-Code Auszüge</b>	<b>249</b>
<b>Abkürzungsverzeichnis und Akronyme</b>	<b>253</b>
<b>Formelzeichen und Variablen</b>	<b>257</b>
<b>Tabellenverzeichnis</b>	<b>259</b>
<b>Abbildungsverzeichnis</b>	<b>261</b>
<b>Literaturverzeichnis</b>	<b>269</b>



# 1. Einleitung

Bereits im Jahr 2004 zeigte der *Toyota Prius* mit seinem Multi-Information Display den Trend der Displays im Automobil an. Es handelte sich um ein Liquid Crystal Display (LCD, Flüssigkristallanzeige) mit Touch [Toy04, S. 146]. Durch die Fortschritte in der Display-Technik kommen heutzutage immer mehr Displays im Automobil zum Einsatz. Im Jahr 2022 werden im Durchschnitt  $\approx 1,66$  Displays pro Fahrzeug und im Jahr 2027 etwa 2 Displays pro Fahrzeug erwartet (gerechnet auf die Fahrzeugabsätze) [Cum20a]. Bei dem Großteil der verwendeten Anzeigen handelt es sich um LCDs. Einen kleinen Anteil stellen Displays dar, welche aus selbst-emittierenden Organic Light-Emitting Diode (OLED) Bildpunkten bestehen. Die Bildpunkte werden auch Pixel<sup>1</sup> genannt.

Insbesondere wird zwischen einem Instrument Cluster Display (ICD, Kombi-Instrument Anzeige) und Center Information Display (CID, Zentralanzeige) unterschieden. Erstere sind für den Fahrer relevant und müssen sicherheitsrelevante Information zuverlässig anzeigen. Letztere sind typischerweise für die Darstellung von zusätzlicher, nicht sicherheitsrelevanter Information wie der Navigationskarte oder die Kontrolle des Entertainments zuständig. In Abbildung 1.1 sind solche Displays abgebildet. Der Trend der zunehmenden



(a) Beispiel eines ICDs der Firma *Tianma*. Foto von der *electronic displays Conference 2020*.



(b) Beispiel eines CIDs. Entnommen aus [BHT18].

Abbildung 1.1.: Beispiele verschiedener Displays im Automobil.

Displaygrößen ist in Abbildung 1.2 deutlich gezeigt. Links für die ICDs und rechts für die CIDs. Die Displays werden in den nächsten Jahren im Durchschnitt größer.

Zudem wurde vorhergesagt, dass der Markt für ICDs von in den Jahren 2019 bis 2025 mit einem CAGR<sup>2</sup> von 20 % wächst. Eine Umsatzsteigerung von 2 Milliarden USD auf 7 Milliarden USD ist zu erwarten [WY19]. Aufgrund der *COVID-19-Pandemie* sind die Absatzzahlen der automotive Displays im Jahr 2020 zum Vorjahr jedoch um 16 % gefallen. Trotz diesen Umständen soll der Absatz zwischen 2020 bis 2028 von 111

<sup>1</sup>Kunstwort aus *picture* und *element* (Pixel)

<sup>2</sup>Compound Annual Growth Rate

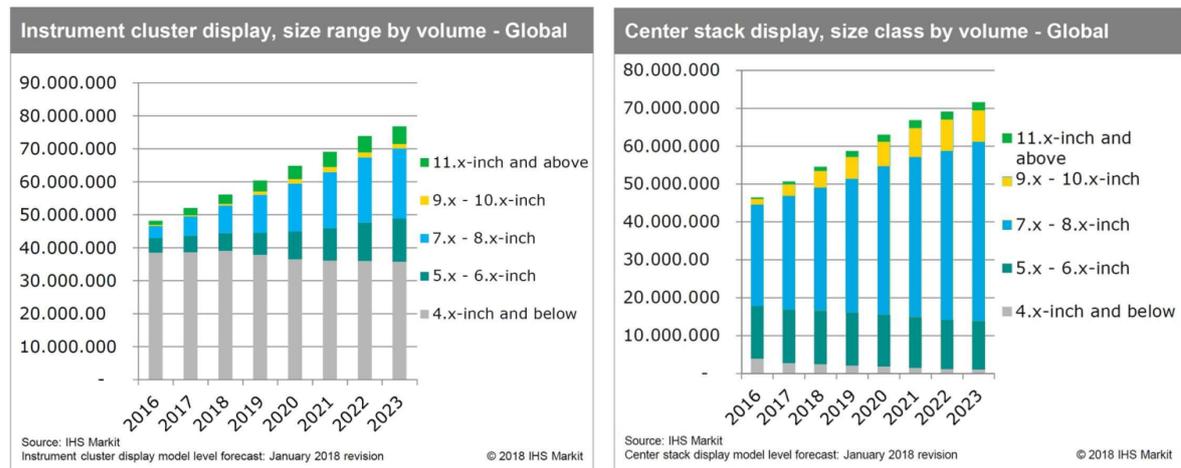


Abbildung 1.2.: Entwicklung des Absatzes von ICDs und CIDs. Entnommen aus [Mar18].

Millionen auf 200 Millionen wachsen [Bla20; Bla21]. Die Hauptursache für den Absatzrückgang in 2020 ist der vorherrschende Chipmangel [Kam21].

## 1.1. Motivation und Hinführung zum Thema

Wie oben erwähnt, wachsen der automotiv Displaymarkt und Absatzzahlen zukünftig weiter. Die Bildqualität wird immer wichtiger und der Wunsch nach höherer Spitzenleuchtdichte und hohen Kontrasten nimmt stetig zu. Zudem führen die immer höheren Displayauflösungen zu einem höheren Leistungsbedarf um dieselbe Leuchtdichte liefern zu können, wie mit einem niedrig-auflösenden Display. Die Ursache liegt in der für das Licht transparenten Fläche (*Aperture-Ratio*), die bei gleicher Displaygröße aber steigender Auflösung abnimmt. Die Ablesbarkeit unter Sonneneinstrahlung soll ebenfalls verbessert werden [Bla18]. Zudem muss ein Display im Automobil sicher und robust sein und einen geringen Leistungsumsatz vorweisen. Ein integriertes Interieur-Design ausgestattet mit Displays von hoher Bildqualität und speziellen Features kann einem Fahrzeug ein Unique Selling Point (USP, Alleinstellungsmerkmal) verleihen. Künftige Konzepte und Displaytrends sind in Abbildung 1.3 beispielhaft dargestellt.



Abbildung 1.3.: Zukünftige Entwicklung im Cockpit (oben BMW) und Interieur (unten Daimler). Entnommen aus [Bla18].

## 1.1. Motivation und Hinführung zum Thema

In der heutigen Konsumgesellschaft wird ein Smartphone bei einem Durchschnittsalter von 2,75 Jahren ersetzt [Dan21]. Im Jahr 2019 betrug das Durchschnittsalter eines Pkw hingegen 9,5 Jahre (siehe Abbildung 1.4). Damit wird schnell deutlich, dass auch die Komponenten wie Displays für eine sehr lange Laufzeit optimiert werden müssen, ohne dass das Bild mit der Zeit degradiert wird. Hier müssen also die Lebensdauer sowie die Zuverlässigkeit entsprechend hoch sein. OLED Displays liefern ein *perfektes* Schwarz und einen großen Farbraum, leiden jedoch unter Alterungseffekten und sind deutlich teurer als LCDs. Zudem muss eine hohe Leuchtdichte vermieden werden, um den Alterungsvorgang nicht zu beschleunigen. Aus diesem Grund

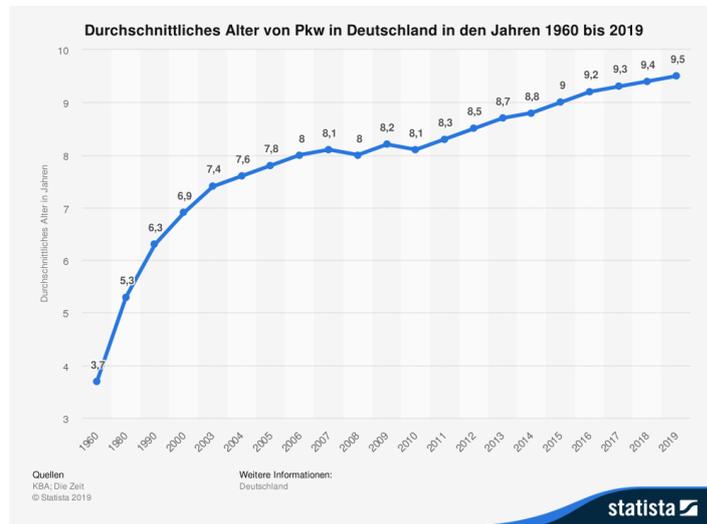


Abbildung 1.4.: Durchschnittliches Alter von Pkws in Deutschland. Entnommen aus [KZ19].

stellen die Mehrzahl der Displays im Automobil Flüssigkristalldisplays dar. Eine weitere Anforderung ist die Effizienz bei der Bilddarstellung. Bei OLED-Displays wird nur Leistung in den Pixeln umgesetzt, welche Licht emittieren. Ein dunkles Bild setzt folglich auch weniger Leistung um als ein helles Bild. Bei LCDs ist der Leistungsumsatz bisher weitestgehend *unabhängig* vom Bildinhalt. Aufgrund des Funktionsprinzips müssen die Pixel *beleuchtet* werden und der Flüssigkristall sorgt dafür, dass das Pixel in seiner Helligkeit moduliert wird. Das Licht zur *Beleuchtung* der Pixelmatrix wird als *Backlight* oder auch *Hintergrundbeleuchtung* bezeichnet und wird von der Backlight-Unit (BLU) bereitgestellt. Bei den Leuchtmitteln handelt es sich heutzutage um weiße Light-Emitting Diodes (LEDs). Die Phosphorkonversionsschicht ist hierfür direkt auf dem Die (LED-Chip) aufgetragen.

Ist ein Pixel dunkel, so wird dieses trotzdem beleuchtet. Das hat zur Folge, dass auch bei einem sehr dunklen Bild der Leistungsumsatz gleich hoch bleibt. Die BLU setzt ca. 90 % des Strombedarfs des gesamten Displays um [HLL08]. Bei einem weißen Bild kommt von dem emittierten Licht der LEDs beim Betrachter nur etwa 7 % an [Hd09]. Dieser Wert kann jedoch in Abhängigkeit des optischen Stacks etwas variieren. Allein die Farbfilter der Subpixel blockieren  $\frac{2}{3}$  des Lichtes (unter Verwendung von 3 Farbfiltern für RGB).

Aufgrund der fortschreitenden Entwicklung hin zur Elektromobilität und Umweltschutz ist ein hoher Leistungsbedarf kontraproduktiv. Die Europäische Union legte einen CO<sub>2</sub> Grenzwert von 130 g km<sup>-1</sup> im Jahr 2015 fest. Für 2021 wurde er auf 95 g km<sup>-1</sup> gesenkt (Flottenmittelwert) [Koe20]. In Abbildung 1.5 sind die CO<sub>2</sub> Grenzwerte bis 2030 abgebildet. Somit ist klar, dass auch Displays im Automobil effizienter werden müssen. Aufgrund des Preises von OLED Displays und der Lebensdauer Problematik dienen sie als Technologietreiber bei der Weiterentwicklung von LCDs in Richtung hoher Bildqualität und Leistungseffizienz.

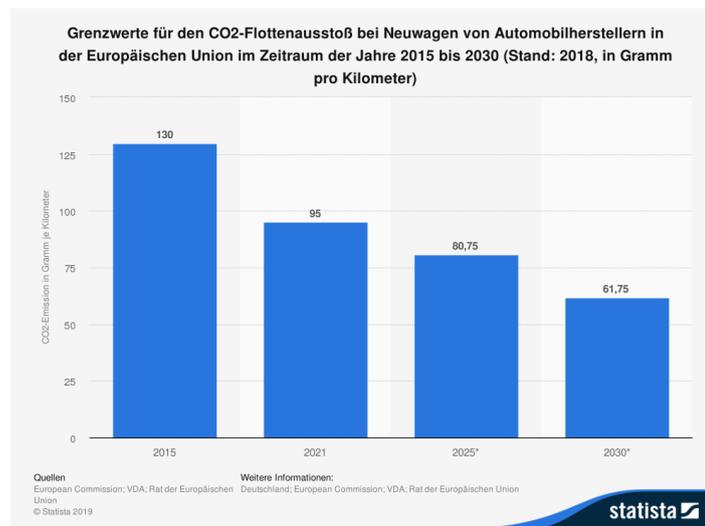


Abbildung 1.5.: EU Grenzwerte für den CO<sub>2</sub> Ausstoß bei Neuwagen. Entnommen aus [Han18].

Da für die Erzeugung der Hintergrundbeleuchtung LEDs zum Einsatz kommen, können diese auch *gedimmt* werden. Typischerweise wird mit Pulse Width Modulation (PWM, Pulsweitenmodulation) mit Konstantstrom die Photonenemission eingestellt [Ana08; Sun12]. Dadurch lässt sich ein Dimming-Verfahren nutzen, um die Leuchtdichte an den Bildinhalt anzupassen. So ein Verfahren in Kombination mit einer Pixelkompensation (Erhöhung der Transmission, um die Zielleuchtdichte zu erhalten) wird als Global-Dimming bezeichnet, wenn alle LEDs mit ein und demselben PWM Duty-Cycle betrieben werden. Können die LEDs auch unabhängig voneinander gesteuert werden, wird von einem Local-Dimming Verfahren gesprochen. Die Berechnung der optimalen LED-Werte stellt eine Herausforderung für die Echtzeit-Prozessierung dar, wie später noch detaillierter gezeigt wird.

Aufgrund der Kosten und Bautiefe haben sich zunächst sogenannte Edge-Lit LCDs als ICDs durchgesetzt. Die LEDs sitzen hier an einer der Displaykanten. Das Licht wird durch *Light-Guides* hinter das LC-Panel geleitet und beleuchtet so die Pixel-Matrix. Hierfür wurde bereits der Sorted Sector Covering (SSC) Local-Dimming-Algorithmus, welcher am Lehrstuhl für Mikroelektronik an der Universität des Saarlandes (LME) entwickelt wurde, erfolgreich angewandt. In einem späteren Kapitel wird darauf eingegangen. Solch ein Display hat jedoch den Nachteil, dass die Hintergrundbeleuchtung aufgrund ihrer Leuchtdichteverteilung nicht *lokal* genug eingestellt werden kann. Befinden sich die LEDs in einer Matrix-Anordnung hinter dem LC-Panel, wird ein solches Matrix-Backlight in Fachkreisen häufig Direct-Lit genannt. Diese Bezeichnung wird im weiteren Verlauf dieser Arbeit als Synonym für Matrix-Backlight verwendet. Solch eine Anordnung erlaubt eine *lokale* Ausleuchtung und somit eine bessere Adaption an das anzuzeigende Bild [Shi+06]. Aufgrund der benötigten Treiber und einer höheren LED-Anzahl ist eine Direct-Lit BLU teurer als Edge-Lit. Der Trend zu Direct-Lit wird beobachtet, da durch eine höhere Lokalität das Resultat zu einem besseren Kontrast führt. Zudem kann eine Optimierung der Leuchtdichteverteilung zur Beleuchtung der Pixel das Local-Dimming System leistungseffizienter machen. In dieser Arbeit wird auch darauf eingegangen.

Der Bedarf an *hochauflösenden* BLUs in Kombination mit Local-Dimming ist vorhanden und zeigt sich durch das Interesse vieler Original Equipment Manufacturers (OEMs, Originalausrüstungshersteller) sowie Zulieferer. Aufgrund der schnellen Entwicklung muss der Kompetenzaufbau bei den Zulieferern schnell erfolgen, um relevante neue Technologien identifizieren zu können und den OEMs entsprechende Lösungen

anbieten zu können. Folglich sind hohe Investitionen in Forschung und Entwicklung notwendig [Mog+19] um weiterhin wertschöpfend tätig zu bleiben.

Im Rahmen dieser Arbeit wurden unter anderem Projekte in Zusammenarbeit mit *Continental*, *Nichia* und auch *BMW* zum Thema Local-Dimming durchgeführt. Teilweise wurde benötigte Hardware wie die BLUs von den Projektpartnern gestellt. Mithilfe dieser Hardware konnten im Rahmen der Arbeit wissenschaftliche Problemstellungen untersucht und bearbeitet werden.

Neben den genannten Anforderungen an Displays bezüglich der Qualität und Effizienz, stellen die immer weiter steigenden LED-Zahlen eine Herausforderung für das bisherige SSC Local-Dimming dar, wie im Verlauf dieser Arbeit gezeigt wird. Der Einsatz von einem hochauflösenden Matrix-Backlight ist unumgänglich, um in naher Zukunft die Bildqualität nahe von OLED zu erreichen und eine Qualitätssteigerung sowie ein USP im Automobil zu erzielen. Das Potential der höher aufgelösten BLUs ist sehr hoch und verspricht deutlich höheres Powersaving bei gesteigerter Qualität. Neue Methoden sind unabdingbar, da neben dem Local-Dimming Algorithmus auch das BLU-Design sowie Hardwarekosten berücksichtigt werden müssen, um ein effizientes Displaysystem zu erhalten. Im Rahmen dieser Arbeit wird gezeigt, dass das BLU Design einen hohen Einfluss auf die mögliche Leistungseinsparung und Artefakte hat. Wird dabei der Local-Dimming Algorithmus nicht berücksichtigt, so wird das Potential an Bildqualität und Leistungseinsparung nicht ausgeschöpft. Somit entstehen einige Interdependenzen zwischen der BLU und dem Algorithmus.

## 1.2. Zielsetzung und Struktur der Arbeit

In dieser Monografie mit dem Titel

*„Effizienter Local-Dimming Algorithmus und Matrix-Backlight-Designoptimierung für LC-Displays zur Steigerung der Energie-Effizienz und der Bildqualität mit Fokus auf automotive-spezifische Anforderungen“*

werden die genannten Herausforderungen näher analysiert und neue Methoden entwickelt, um diese zu meistern. Auch der Einfluss der Leuchtdichteverteilung auf die Performance des Local-Dimming Algorithmus wird untersucht und Designvorschläge für BLUs in Bezug auf die Leuchtdichteverteilung abgeleitet. Das Ziel ist ein Local-Dimming System, bei dem die einzelnen Komponenten wie der Algorithmus und BLU so aufeinander abgestimmt sind, dass die resultierende Bildqualität verbessert, die Effizienz gesteigert, sowie die Kosten gesenkt werden können.

Für den im Rahmen dieser Arbeit entwickelten Local-Dimming Algorithmus gelten zeitliche Constraints (Bildwiederholrate von 60 Hz) und weitere limitierende Faktoren wie steigender Speicherbedarf und somit steigende Kosten des Static Random Access Memorys (SRAMs). Zudem führt die steigende Auflösung des LC-Panels zu einer deutlich höheren Datenrate (Bilddatenstrom). Eine Bedingung für den Local-Dimming Algorithmus ist die Implementierung direkt auf der *Pixelpipeline* ohne Verwendung von Frame-Buffern. Dies führt zur leichteren Zertifizierung nach Automotive Safety Integrity Level (ASIL) und erlaubt Echtzeit-Verarbeitung. Anders als bei bisher veröffentlichter Literatur, wird hier der sogenannte *Crosstalk* zwischen den LEDs beachtet, um eine am Optimum liegende Lösung zu berechnen. Dazu wird das Local-Dimming als mathematisches Optimierungsproblem definiert, indem die zugrundeliegenden physikalischen Gegebenheiten abgebildet werden. Um den Crosstalk zu berücksichtigen, wird das Problem mit einem iterativen Ansatz

gelöst. Zudem sind qualitätssichernde Features zu erarbeiten und umzusetzen.

Zusammengefasst hat diese Arbeit zum Ziel die Auswirkungen verschiedener Leuchtdichteverteilungen der LEDs bei hochauflösenden Matrix-BLU auf die Performance des Local-Dimmings zu untersuchen und Vorschläge für optimierte BLU-Designs zu liefern. Es wird unter anderem auf die Full Width at Half Maximum (FWHM, Halbwertsbreite) der Leuchtdichteverteilung eingegangen, sowie die Auswirkungen der LED-Anzahl in einer BLU auf die Local-Dimming Resultate untersucht. Das daraus abgeleitete Verhältnis zwischen dem FWHM und dem LED-Pitch stellt einen Kompromiss zwischen der Local-Dimming Performance und der Kosten dar, wie im Rahmen dieser Arbeit gezeigt wird. Es wird festgestellt, dass die Hardwarekosten mit der Herangehensweise des State-of-the-art (SoA, Stand der Technik) SSC Local-Dimming Algorithmus bei hochauflösenden BLUs schnell sehr hoch werden und nicht mehr tragbar sind. Zudem lassen sich aufgrund der hohen LED-Zahl die Berechnungen nicht mehr innerhalb eines Frames realisieren. Aus diesem Grund werden neuartige Methoden vorgestellt, um die Hardwarekosten gering zu halten und den Anforderungen für automobile Displays zu genügen. Dabei soll die Performance des Local-Dimming Algorithmus weiter erhöht werden. Das Ziel ist die Wettbewerbsfähigkeit der LCDs mit der Bildqualität der OLED Displays sicherzustellen. Dazu wird ein modellbasierter Ansatz vorgestellt, welcher auf Modellen des Beitrages des Lichts pro Pixel basiert. Durch diese Methode ist es anders als in SoA nicht mehr notwendig, die Lichtbeiträge jeder einzelnen LED zu speichern, da nun eine bestimmte ausgewählte Menge an LEDs zur Repräsentation aller LEDs genutzt werden kann. Aufbauend werden Konzepte zur Qualitätssteigerung entwickelt.

Die Methoden und Algorithmen werden für den Proof-of-Concept zunächst in MATrix LABoratory (MATLAB) umgesetzt. Dieses modellierte Verhalten in MATLAB dient aufbauend zur Validierung als Referenz für die Implementierung auf einem Field Programmable Gate Array (FPGA, Feld programmierbare Gatter-Anordnung) mithilfe der Hardware Description Language (HDL, Hardwarebeschreibungssprache) *Verilog*.

Es ist inzwischen eine gängige Praxis in der Chip-Industrie, dass ein Algorithmus in HDL an Nutzer (Lizenznehmer) übertragen und so verwendet bzw. verwertet wird. Er wird auch als Intellectual Property (IP) bezeichnet. Diese Arbeit verfolgt unter anderem auch das Ziel, dass die hier erarbeitete Technologie so hochwertig, adaptierbar und effizient ist, dass sie als Local-Dimming IP an Nutzer übertragen werden kann. Das IP bezeichnet im Folgenden auch den Local-Dimming Algorithmus in HDL-Design. Sie ist in Demonstratoren umzusetzen und zu validieren. Die Tauglichkeit und Verwertbarkeit der neuen Methoden und des Gesamtsystems im Automobil soll dabei demonstriert werden.

Die Randbedingungen bei der Entwicklung lassen sich nach Gessler [Ges14] in *technische*, *system-technische* sowie *ökonomische* Randbedingungen unterteilen. Für das hier entwickelte Local-Dimming System kann wie folgt unterteilt werden.

### **Technisch**

Der Algorithmus muss für eine hohe Anzahl von LEDs in der BLU funktionieren und auch hardwareeffizient sein. Zunehmende Datenraten (des Bilddatenstromes) durch höhere Auflösung bedingt, müssen unter Einhaltung des zeitlichen Constraints bzw. Limits von *einem* Frame verarbeitet werden können. Zudem muss das System eine geringe Latenz zwischen den Bildeingangs- und Ausgangsdaten aufweisen. Diese soll deutlich

unterhalb der Wahrnehmung liegen (Echtzeit). Der gesamte Ressourcenbedarf des Local-Dimming-IPs soll einen Bruchteil eines Timing-Controller (TCON)-Chips, auf dem das IP integriert wird, nicht übersteigen.

### **System-Technisch**

Als system-technische Randbedingung ist der Verzicht auf Frame-Buffer sehr wichtig (auch für Latenz und Echtzeit). Neben den geringeren Kosten führt ein Ausfall des Local-Dimming Systems zu einem schwarzen Bild. Der Fahrer kann nicht durch einen Frame-Freeze, welcher durch einen Random Access Memory (RAM) produziert wird, fälschlicherweise getäuscht werden (e. g. falsche Geschwindigkeitsanzeige wegen *Standbild*). Zudem soll die Flexibilität, also die Anpassungsfähigkeit an verschiedene BLUs sowie LC-Panels (Seitenverhältnisse und Auflösung) gegeben sein. Auch eine Parametrisierung ist erforderlich, um den Beziehern der IP die Möglichkeit zu bieten, den Algorithmus an ihre Wünsche anzupassen. Beispielsweise kann ein Trade-Off zwischen Bildqualität und Leistungseinsparung ausbalanciert werden. Eine gute Anwendbarkeit im Serienprodukt ist anzustreben.

### **Ökonomisch**

Die Entwicklungsdauer der Local-Dimming IP bis zur Anwendung im Demonstrator bzw. bis zur Markteinführung muss aus wirtschaftlicher Sicht kurz gehalten werden. Eine Realisierung im FPGA mit Hinblick auf eine mögliche spätere Überführung in ein Application-Specific Integrated Circuit (ASIC, Anwendungsspezifische integrierte Schaltung), wie beispielsweise einen TCON, bietet einen Zeit- und Kostenvorteil. Außerdem ist der Einsatz von FPGAs in Kleinserien sinnvoll. Wird jedoch die Nutzungsmöglichkeit auf weitere Displays (e. g. weitere Pkw Modelle) hoch-skaliert, so können die Kosten durch eine ASIC/TCON-Implementierung signifikant gesenkt werden (hier ist auch die *system-technische* Randbedingung der Flexibilität relevant). Die Nutzung neuerer Integrated Circuit (IC, Integrierter Schaltkreis)-Fertigungsprozesse und kleinere Strukturbreiten erlauben höhere Frequenzen und auch eine höhere Die-Ausbeute, da die Chip-Fläche kleiner wird [Mur64; IC 19].

### **Struktureller Aufbau der Arbeit**

Im Rahmen dieser Arbeit wurden Zwischenergebnisse und Resultate neben einer inländischen Konferenz, der *electronic displays Conference* [Sch+16b] auch auf internationalen Konferenzen in den USA, Japan, Südkorea sowie Europa (den Niederlanden) veröffentlicht [SSX15; Sch+17; Sch+18; SRX19; SX19; SAX20] und präsentiert. Zusammengefasst zählen dazu die *SID<sup>3</sup> Displayweek*, die *SID Vehicle Displays and Interfaces*, die *International Display Workshops (IDW)*, das *International Meeting on Information Display (IMID)* sowie das *Mid-Europe Chapter* der *SID*. Ein Journal Paper wurde im *Journal of Information Display* publiziert [Sch+19b]. Daneben besteht eine Ko-Autorenschaft bei den Veröffentlichungen [Grü+16; Xu+18a; Grü+19; XAS22] sowie in der *Information Display* [Xu+18b], welche einen hohen *Impact Factor* aufweist. Zudem wurde ein Patent zum Thema Global-Dimming angemeldet [SSX16].

Neben dem Literaturverzeichnis ab Seite 269, werden einige Quellen wie Master- oder Bachelorarbeiten im Abschnitt *Graue Literatur* auf Seite 285 separat gelistet. Diese Quellen sind nur bedingt zitierfähig, da sie nicht für jedermann einsehbar sind. Sie sind jedoch über die interne Bibliothek des LME auf Anfrage abrufbar.

---

<sup>3</sup>SID: The Society for Information Display

In Abbildung 1.6 ist der strukturelle Aufbau dieser Arbeit dargestellt. Dabei ist der Lesefluss mit roten Pfeilen, sowie die Erkenntnisse und Abhängigkeiten mit schwarzen Pfeilen angedeutet. Die Kapitel, welche einen großen Teil der Eigenleistung und neuartigen Methoden enthalten, sind grün hinterlegt.

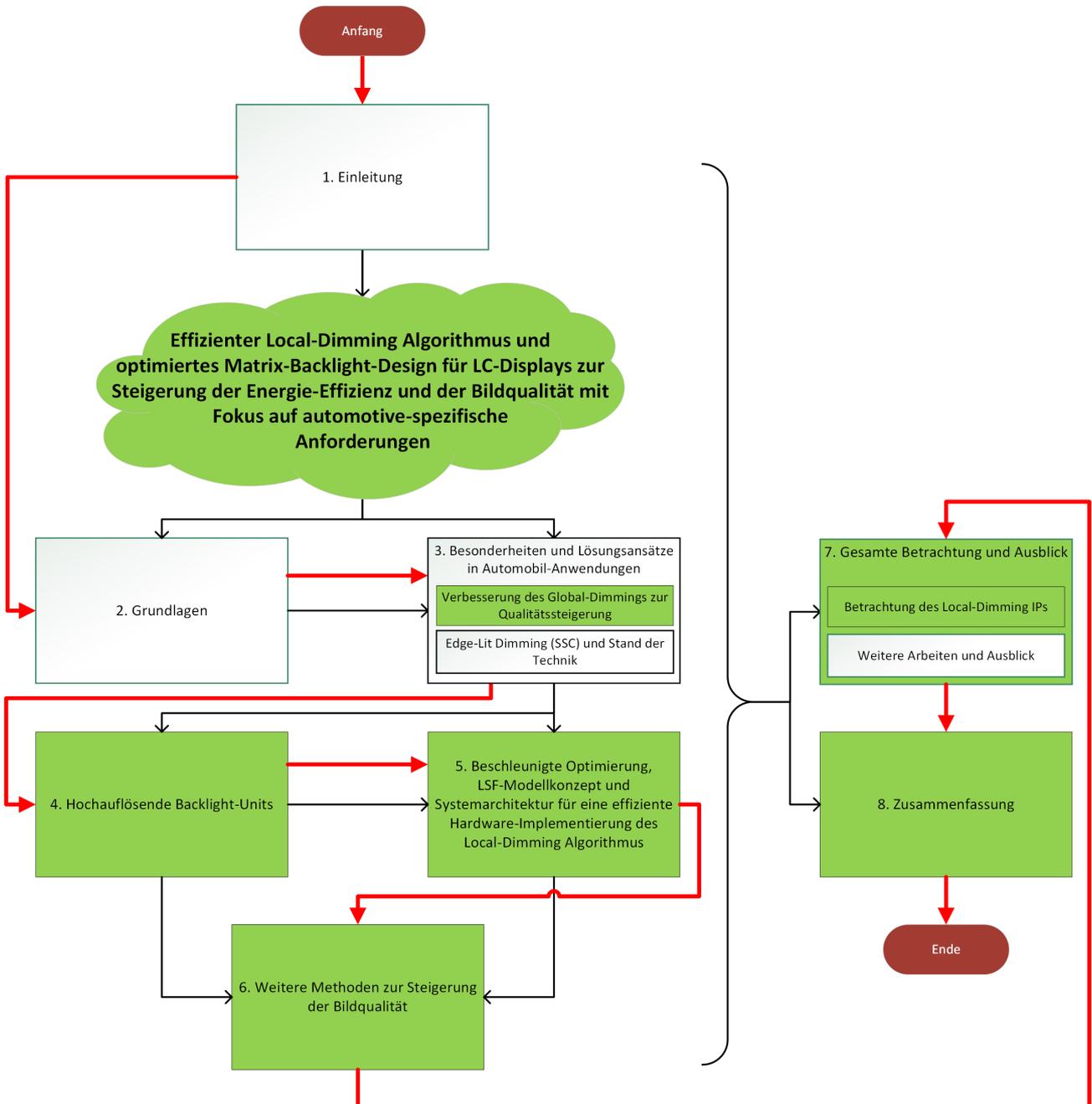


Abbildung 1.6.: Struktur dieser Monografie. Der Lesefluss ist in Rot und die strukturellen Abhängigkeiten und der Erkenntnisflow sind mit schwarzen Pfeilen dargestellt. Grün hinterlegt sind die Kapitel, welche hauptsächlich eine Eigenleistung im Rahmen dieser Arbeit darstellen.

Die Einführung und Motivation für die Arbeit

*Effizienter Local-Dimming Algorithmus und Matrix-Backlight-Designoptimierung für LC-Displays zur Steigerung der Energie-Effizienz und der Bildqualität mit Fokus auf automotive-spezifische Anforderungen*

wurde im Kapitel 1 in der „Einleitung“ gegeben. Zunächst werden in Kapitel 2 neben den Grundlagen der menschlichen Wahrnehmung, die Funktionsweisen der LCDs sowie OLED Displays erläutert. Es wird auch auf die Imperfektionen von LCDs eingegangen sowie die Grundlagen des Dimmings vorgestellt. Da die in dieser Monografie erarbeiteten Methoden final durch ein echtes Hardwaredesign auf einem FPGA realisiert werden, wird auch deren Funktionsweise beschrieben und der topologische Aufbau eines LCD-Moduls vorgestellt.

Kapitel 3 „Besonderheiten und Lösungsansätze in Automobil-Anwendungen“ behandelt die Anforderungen von Displays im Automobil. Für einen Low-Cost-Ansatz werden neue Methoden vorgestellt, um bei einem LCD ein Global-Dimming System zu realisieren. Dazu wird ein verbesserter Global-Dimming Algorithmus speziell für die Automobilanwendung entwickelt und auf Tauglichkeit untersucht. Teile dieses Global-Dimming Systems wurden in eine Lehrveranstaltung *Praktikum Mikroelektronik* in vereinfachter Weise integriert, um den Studenten die aktuellen Probleme im LCD-Bereich methodisch näherzubringen und die Grundlagen einer HDL zu erlernen. Dabei wird ein Global-Dimming System basierend auf einer Pixelpipeline von den Studenten implementiert und auf einem FPGA und Display validiert.

In diesem Kapitel wird zudem der Stand-der-Technik des am LME entwickelten SSC Local-Dimming Algorithmus für BLUs mit einer geringen LED-Anzahl vorgestellt (Abschnitt 3.5). Dabei handelt es sich typischerweise um Edge-Lit-Geräte.

Kapitel 4 „Hochauflösende Backlight-Units“ hat die neuen, hochauflösenden BLUs zum Thema. Es wird untersucht, wie sich der veränderte Einfluss, also der Lichtbeitrag einer LED in einem Pixel auf den Local-Dimming Algorithmus auswirkt. Dabei werden Qualitätsmetriken sowie die Modellierung von Leuchtdichteverteilungen behandelt. Ein Ziel ist die Ermittlung einer charakteristischen Größe für das BLU Design durch das Verhältnis des FWHM einer Leuchtdichteverteilung zum LED-Pitch sowie einen Vorschlag eines abgeleiteten Trade-Offs zu liefern. Aufbauend werden verschiedene Parameter der Leuchtdichteverteilung auf die Robustheit des Gesamtsystems untersucht.

Kapitel 5 „Beschleunigte Optimierung, LSF-Modellkonzept und Systemarchitektur für eine effiziente Hardware-Implementierung des Local-Dimming Algorithmus“ hat zum Ziel, neue Methoden und Konzepte zu erarbeiten, um eine effiziente und hardware-schonende Berechnung der LED-Werte für ein Local-Dimming System zu ermöglichen. Die Auswirkungen und der Hardwareaufwand einer gestiegenen LED-Anzahl in der BLU werden zunächst aufgezeigt. Danach wird das Optimierungsziel beschrieben und die Entwicklung einer modellbasierten Nutzung der Lichtbeitragsinformation durchgeführt. Die daraus resultierende notwendige Neukonzeption der *Pixelorder* wird ebenfalls diskutiert und schließlich als FPGA Design basierend auf der Pixelpipeline umgesetzt.

In Kapitel 6 „Weitere Methoden zur Steigerung der Bildqualität“ werden basierend auf dem System aus Kapitel 5 weitere neuartige Methoden entwickelt, um die Bildqualität spezifisch für Automobile-Anwendungen weiter zu steigern. Insbesondere das sogenannte *Halo*-Artefakt soll entschärft werden. Es werden algorithmenseitige Methoden vorgeschlagen, um den Schwarzwert an den Displayrändern weiter zu verbessern, um eine bessere Integration des Displays zu ermöglichen. Für den für die Automobil-Anwendung wichtigen

Sicherheitsaspekt des LED-Ausfalls wird eine Lösung präsentiert und weitere Methoden zur Abdeckung noch höherer Matrix-Backlight-Auflösungen vorgestellt.

Kapitel 7 „Gesamte Betrachtung und Ausblick“ betrachtet das hier entwickelte IP als Gesamtsystem und gibt einen Ausblick über aufbauende Themen und weitere Entwicklungen.

Zum Abschluss der Arbeit wird in Kapitel 8 diese Monografie zusammengefasst sowie die wichtigen Erkenntnisse und Ergebnisse dieser Arbeit resümiert. Als Beispiel für innovatives Potential der in dieser Arbeit entwickelten Technologie wird eine automotive-taugliche Anwendung in einem sogenannten *Shy Tech* Display präsentiert.

Zusätzlich befinden sich in den Anhänge A und B neben Abbildungen, welche ausgelagert wurden, um den Lesefluss nicht zu stören, auch eine Übersicht der genutzten Testbilder. Im Fließtext wird bei Bedarf darauf verwiesen.

Im nächsten Kapitel werden die für diese Arbeit wichtigen Grundlagen und Funktionsweisen vorgestellt. Aufbauend auf den radio- und photometrischen Grundlagen und Wahrnehmung wird unter anderem auf die Funktionsweise von LCDs und OLED Displays eingegangen sowie die Grundprinzipien des Dimmings eingeführt. Die für das Hardwaredesign notwendigen Grundlagen zum Timing und FPGA Aufbau werden ebenfalls im Abschnitt 2.6 erläutert.

## 2. Grundlagen

In diesem Kapitel werden die notwendigen Grundlagen eingeführt, welche für die Umsetzung und Verständnis der in dieser Arbeit vorgestellten Methoden und Systemen relevant sind. Es wird neben der Funktionsweise von Displays auf die menschliche Wahrnehmung das Human Visual System (HVS, menschliche visuelle System) als auch auf die Grundlagen der Dimming-Verfahren vorgestellt, welche die Grundgerüste dieser Arbeit bilden. Zunächst wird auf die Radiometrie und Fotometrie eingegangen.

### 2.1. Radio- und Fotometrische Grundlagen

Im Rahmen dieser Arbeit wurden verschiedene Messungen durchgeführt. Zum Verständnis dieser Messungen und der Messgrößen wird in diesem Abschnitt ein Überblick über die verschiedenen Terminologien und Einheiten gegeben. Der Begriff *Licht* bezeichnet den Teil der für Menschen sichtbaren Strahlung des elektromagnetischen Spektrums. Nach [DIN17a, Teil 7] wird hierbei der Wellenlängenbereich zwischen 380 nm und 780 nm definiert. Dieser Bereich ist somit auch für Displays relevant. Doch auch angrenzende Bereiche im UV-Bereich und Infrarot werden zusammen mit dem genannten Bereich als *optische Strahlung* bezeichnet. In Abbildung 2.1 ist das elektromagnetische Spektrum abgebildet und zeigt auch den sichtbaren Teil mit seiner Einordnung an.

Das Ziel der Radiometrie (Strahlungsphysik) ist die Erfassung der Strahlungsenergie [Poy03]. Ein Spektrometer kann dabei für den Bereich des sichtbaren Lichts eingesetzt werden die Strahlungsenergie in Abhängigkeit von der Wellenlänge zu messen. Die Auflösung in der Wellenlänge (also das schmalste Wellenlängenband) hängt vom genutzten Messgerät ab. Gute Ergebnisse ergeben sich bei einer Schrittweite von unter 5 nm.

Der weitere Teil dieses Kapitels basiert größtenteils auf den Inhalten von [Ped+02] und [Poy03]. Mit dem Planckschen Wirkungsquantum  $h$  und der Lichtgeschwindigkeit  $c$  ergibt sich die wellenlängenabhängige Photonenenergie  $Q_e$  zu

$$Q_e = h \cdot \frac{c}{\lambda} = h \cdot f \quad (2.1)$$

Der *Strahlungsfluss* bzw. -leistung ist definiert als [Ped+02]

$$\Phi_e = \frac{dQ_e}{dt} \quad (2.2)$$

Sei  $\phi$  nun der Photonenstrom, welcher die Photonenanzahl pro Zeit definiert, dann gilt für die Strahlungsleistung der betrachteten Frequenzen im Spektrum

$$\Phi_e = h \cdot \int_0^{\infty} \frac{d\phi}{df} \cdot f df \quad \text{mit} \quad [\Phi_e] = W \quad (2.3)$$

Die Strahlungsleistungsdichte bezogen auf eine Empfängerfläche wird als *Bestrahlungsstärke* bezeichnet

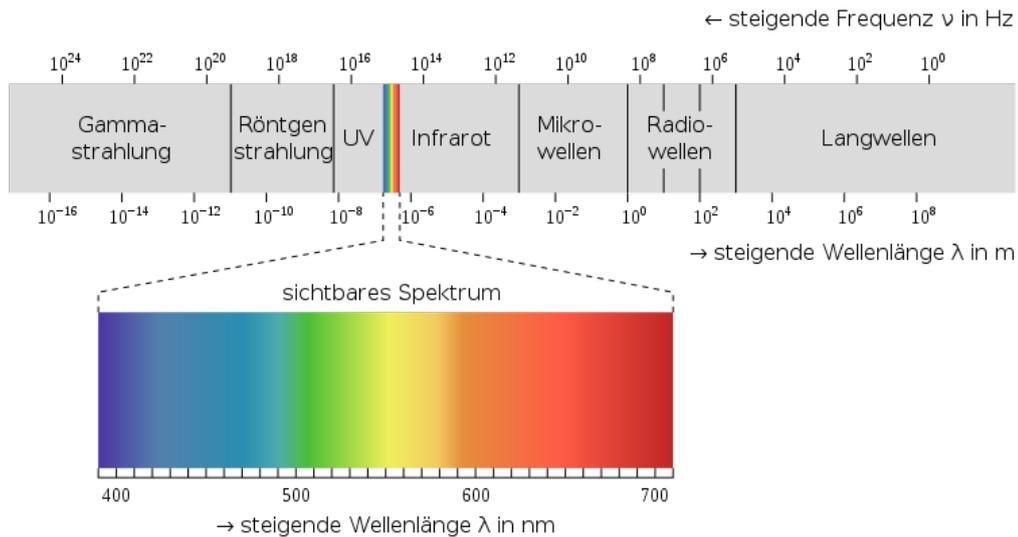


Abbildung 2.1.: Elektromagnetisches Spektrum mit sichtbarem Bereich. Entnommen aus [Mat09].

[Ped+02] und definiert als

$$E_e = \frac{d\Phi_e}{dA} \quad \text{mit} \quad [E_e] = \text{W m}^{-2} \quad (2.4)$$

Eine weitere wichtige Größe ist die sogenannte *Strahlstärke*  $I_e$ . Diese definiert den Strahlungsfluss einer Punktquelle pro Raumwinkel wie folgt [Ped+02]

$$I_e = \frac{d\Phi_e}{d\Omega} \quad \text{mit} \quad [I_e] = \text{W sr}^{-1} \quad (2.5)$$

Mithilfe der Strahlstärke kann die *Strahldichte*  $L_e$  definiert werden. Diese gibt die Strahlstärke  $I_e$  einer Quellenfläche an, welche orthogonal zur Beobachtungsrichtung steht [Ped+02]. Mit Einsetzen von Formel (2.5) wird die Strahldichte wie folgt berechnet

$$L_e = \frac{dI_e}{dA \cdot \cos \epsilon} = \frac{d^2\Phi_e}{d\Omega dA \cdot \cos \epsilon} \quad \text{mit} \quad [L_e] = \text{W sr}^{-1} \text{ m}^{-2} \quad (2.6)$$

Der Winkel  $\epsilon$  ist hierbei der Winkel zwischen der Flächennormalen und der Beobachtungsrichtung.  $A \cdot \cos \epsilon$  stellt also eine Projektion der Fläche dar. Sie wird in [Ped+02] als „effektive Senderfläche“ bezeichnet.

Diese vorgestellten Größen mit dem Suffix  $e$  stellen radiometrische Größen dar, während der Suffix  $v$  für die fotometrischen Größen steht. Anders als die Radiometrie, berücksichtigen die lichttechnischen Größen (*Fotometrie*) die Empfindlichkeit des Auges auf die verschiedenen Wellenlängen. Die standardisierte spektrale *Hellempfindlichkeitskurve* (V-Lambda-Kurve) wird in Abschnitt 2.2 vorgestellt. Sie wird als Funktion  $V(\lambda)$  beschrieben. Diese V-Lambda Kurve wird hierbei mit dem Maximalwert des fotometrischen Strahlungsäquivalents  $K_m$  skaliert und es ergibt sich das wellenlängenabhängige spektrale fotometrische

Strahlungsäquivalent

$$K(\lambda) = K_m \cdot V(\lambda) \quad (2.7)$$

und wird auch als *absolute spektrale Empfindlichkeit* bezeichnet [Ped+02]. Die maximale Empfindlichkeit der menschlichen Wahrnehmung liegt bei 555 nm für das Tagsehen und bei 507 nm für das Nachtsehen [DIN17a, Blatt 3]. Für das Tagsehen bei dieser Wellenlänge resultiert ein Strahlungsfluss von  $\Phi_e = 1 \text{ W}$  in einem  $K_m = 683 \text{ lm W}^{-1}$ .

Somit ergibt sich eine fotometrische Größe aus der radiometrischen Größe durch Multiplikation mit der wellenlängenabhängigen absoluten spektrale Empfindlichkeit. Die analoge Größe zum Strahlungsfluss  $\Phi_e$  ist der Lichtstrom  $\Phi_v$ , welcher sich wie folgt berechnet

$$\Phi_v = \int_{380 \text{ nm}}^{780 \text{ nm}} K(\lambda) \cdot \frac{d\Phi_e(\lambda)}{d\lambda} d\lambda \quad \text{mit } [\Phi_v] = \text{lm (Lumen)} \quad (2.8)$$

Die Bestrahlungsstärke (Formel (2.4)) mündet fotometrische in der *Beleuchtungsstärke* wie folgt:

$$E_v = \frac{d\Phi_v}{dA} \quad \text{mit } [E_v] = \text{lm m}^{-2} = \text{lx (Lux)} \quad (2.9)$$

Die *Lichtstärke*, ist das Analogon zur Dichte des Lichtstromes über einen Raumwinkel (Formel (2.5)) und hat die SI-Einheit *Candela*.

$$I_v = \frac{d\Phi_v}{d\Omega} \quad \text{mit } [I_v] = \text{lm sr}^{-1} = \text{cd} \quad (2.10)$$

Die für diese Arbeit relevanteste fotometrische Größe ist jedoch die *Leuchtdichte* oder im Englischen *Luminance* genannt. In dieser Arbeit wird der Begriff *Luminanz* als Synonym zu Leuchtdichte verwendet. Da ein Display als Flächenstrahler angesehen werden kann, eignet sich die zur Strahldichte (Formel (2.6)) analoge fotometrische Größe

$$L_v = \frac{dI_v}{dA \cdot \cos \epsilon} = \frac{d^2\Phi_v}{d\Omega dA \cdot \cos \epsilon} \quad \text{mit } [L_v] = \text{cd m}^{-2} \quad (2.11)$$

Die Leuchtdichte in  $\text{cd m}^{-2}$  wird im amerikanischen Raum auch als Nit bezeichnet und findet sehr häufige Verwendung in Literatur. Eine Übersicht der radiometrischen und fotometrischen Größen sind in Tabelle 2.1 dargestellt.

### Messgeräte

Zur Durchführung benötigter Leuchtdichtemessungen für diese Arbeit und Projekte wurde der Eldim UMaster [ELD] Kolorimeter genutzt. Sie kann flächige Messungen der Leuchtdichte mithilfe eines Charge-Coupled Device (CCD, ladungsgekoppeltes Bauteil) Sensors durchführen. Dieser Sensor bietet eine 16 Bit Analog zu Digital Umwandlung. Zudem sind verschiedene Farb- sowie ND-Filter auf dem Farbrad der Kamera vorhanden. Auch ein Makroobjektiv kann genutzt werden. Der erwähnte CCD Sensor besteht aus  $4 \cdot 10^6$  Sensorpixel und erlaubt es einen sehr hohen Leuchtdichtebereich in einer Messung abzudecken.

Um neben den Flächenmessungen auch Punktmessungen effektiv durchzuführen, wurde neben dem Kolorimeter noch ein Spektrometer verwendet SpectraScan PR-740 [Pho]. Insbesondere zur automatischen Gamma-Kurvenmessung ist dieser Spektrometer sehr gut geeignet, da er direkt mithilfe von Matlab an-

Strahlungsphysikalische Größen				Lichttechnische Größen		
Benennung	Zeichen	Einheit	Definition	Benennung	Zeichen	Einheit
Strahlungsenergie	$Q_e$	W s		Lichtmenge	$Q$	lm s
Strahlungsfluss	$\Phi_e$	W	$\Phi_e = \frac{dQ_e}{dt}$	Lichtstrom	$\Phi$	lm
Strahlstärke	$I_e$	W/sr	$I_e = \frac{d\Phi_e}{d\Omega}$	Lichtstärke	$I$	cd = lm/sr
Strahldichte	$L_e$	W/(m <sup>2</sup> sr)	$L_e = \frac{dI_e}{dA_1 \cos \varepsilon}$	Leuchtdichte	$L$	cd/m <sup>2</sup>
Bestrahlungsstärke	$E_e$	W/m <sup>2</sup>	$E_e = \frac{d\Phi_e}{dA_2}$	Beleuchtungsstärke	$E$	lx = lm/m <sup>2</sup>
spezifische Ausstrahlung	$M_e$	W/m <sup>2</sup>	$M_e = \frac{d\Phi_{e,H}}{dA_1}$			
Bestrahlung	$H_e$	Ws/m <sup>2</sup>	$H_e = \int E_e dt$	Belichtung	$H$	lx s

Anmerkung:  $\Phi_{e,H}$  ist der Strahlungsfluss einer Strahlungsquelle in den Halbraum,  $dA_1$  = Flächenelement der Quelle,  $dA_2$  = Flächenelement des Empfängers

Tabelle 2.1.: Radiometrische und entsprechende fotometrische Größen in der Übersicht. Entnommen aus [Ped+02].

gesteuert werden kann. Er besitzt 512 Detektorzellen mit jeweils spektralen Auflösung von 0,781 nm. Es sind ebenfalls Commission Internationale de l'Éclairage (CIE, Internationale Beleuchtungskommission) Normbeobachter einstellbar. Neben der reinen Leuchtdichte und Farbortmessung, wird auch eine spektrale Analyse durchgeführt.

Beide Geräte sind in einem Labor (Dunkelraum) untergebracht und lassen sich durch ein Schienenaufhängesystem in der Position anpassen. In Abbildung 2.2 sind die beiden beschriebenen Geräte abgebildet.

## 2.2. Menschliche Wahrnehmung

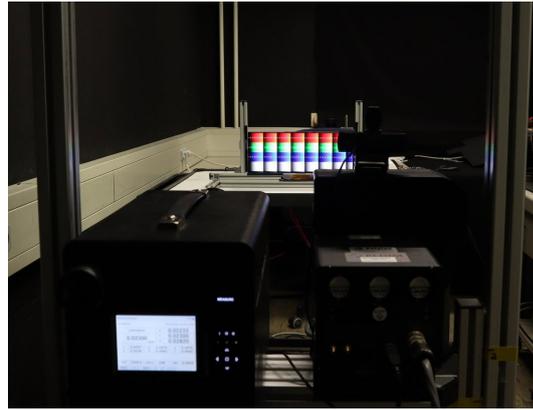
Im vorangegangenen Abschnitt wurde eine Einführung in die Messgrößen der Radio- und Fotometrie gegeben. Nun wird auf die menschliche Wahrnehmung eingegangen, da das Verständnis für die Entwicklung von Algorithmen welche die Bilddarstellung beeinflussen, eine notwendige Bedingung darstellt. Dabei ist diese Kenntnis unabhängig von den zugrunde liegenden Anzeigeräte wie LCDs oder OLED-Displays, notwendig. Diese wird auch als HVS bezeichnet.

### Funktionsprinzip und Anatomie des Auges

Der Aufbau des Auges ist in Abbildung 2.3 abgebildet. Der Weg des einfallenden Lichtes führt durch die Hornhaut und Pupille durch die Linse, welche so akkommodiert, dass durch die Änderung der Brechung des Lichtes, das Quellobjekt scharf auf der Retina (Netzhaut) abbildet [Lev+11]. Dies geschieht mit dem



(a) Frontalansicht auf ELDIM (links) und Spektrometer (rechts). Eine Webcam wird verwendet, um aus dem Kontrollraum das Live-Bild zu sehen und die Ausrichtung zu vereinfachen.



(b) Rückansicht auf beide Messgeräte. Im hinteren Bereich ist ein Display mit Testbild zu sehen, welches vermessen wird.

Abbildung 2.2.: Eldim UMaster und SpectraScan PR-740 im Dunkelraum des LME.

Ziliarmuskel [Lan08]. Auf der Retina selbst befinden sich die Sinneszellen (Photorezeptoren), welche auf die Photonenintensität reagieren. Diese lassen sich in  $125 \cdot 10^6$  Stäbchen und  $7 \cdot 10^6$  Zapfen unterteilen [Ber17]. In Abbildung 2.4 ist die Dichte der Stäbchen (eng. rod) und Zapfen (eng. cone) auf der Netzhaut abgebildet.

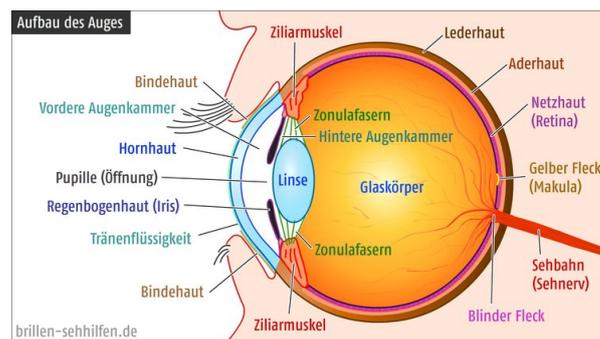


Abbildung 2.3.: Aufbau des menschlichen Auges. Entnommen aus [Auf].

Die Zapfen kommen in drei Varianten vor, den L-, M- und S-Zapfen. Diese sind jeweils auf bestimmte Wellenlängen empfindlich. Die L-Zapfen auf den rot-orangen Frequenzbereich, die M-Zapfen auf den grünen sowie die S-Zapfen auf den blauen Frequenzbereich. Es existieren etwa doppelt so viele L-Zapfen als M-Zapfen und 20 mal mehr M-Zapfen als S-Zapfen. Die Zapfen sind im Bereich des Tagssehens aktiv (*photopisches Sehen*). Die Stäbchen hingegen unterscheiden nicht nach der Wellenlänge der Photonen, sondern nur zwischen hell und dunkel. Diese sind beim Nachtsehen (*skotopisches Sehen*) aktiv. Folglich ist eine Farbwahrnehmung in diesem Bereich nicht möglich. Bei bestimmten Lichtverhältnissen sind beide Sorten der Rezeptoren aktiv (*mesopisches Sehen*) und wird als Dämmerungssehen bezeichnet.

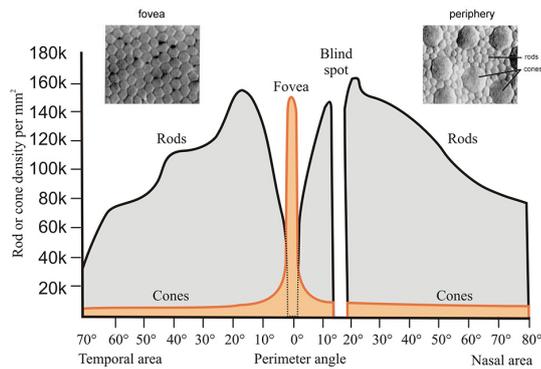


Abbildung 2.4.: Netzhaut mit Dichteverteilung der Zapfen und Stäbchen. Entnommen aus [Gar08].

Diese Bereiche lassen sich für eine Leuchtdichte  $L$  wie folgt unterteilen [Bur21, Anhang 3].

- skotopisches Sehen:  $L < 10^{-3} \text{ cd m}^{-2}$
- photopisches Sehen:  $L > 5 \text{ cd m}^{-2}$
- mesopisches Sehen:  $10^{-3} \text{ cd m}^{-2} < L < 5 \text{ cd m}^{-2}$

Im Bereich des Sehnervs existieren keine Rezeptoren. Dieser Bereich wird auch *blinder Fleck* genannt. Im Gegenteil dazu ist etwa im Zentrum ein Gebiet mit einer sehr hohen Dichte der Zapfen (Makula) und hat zur Folge, dass das einfallende Licht darauf fokussiert wird. Als Folge wird nur in diesem Bereich scharf und auch mit deutlicher Farbdifferenzierung gesehen.

## Helligkeitsempfinden und Tristimulus

Der spektrale Hellempfindlichkeitsgrad auch als *Hellempfindlichkeitskurve* bezeichnet, wurde bereits in Formel (2.7) auf Seite 13 für die Berechnung der spektralen fotometrischen Strahlungsäquivalents verwendet. Die Hellempfindlichkeitskurven für das Tag- und Nachtsehen ( $V(\lambda)$  bzw.  $V'(\lambda)$ ) werden in Abbildung 2.5 abgebildet. Diese Kurven wurden von der CIE experimentell ermittelt.

Gut erkennbar sind die Maxima bei 555 nm für die photopische Wahrnehmung sowie etwa 500 nm-510 nm für das skotopische Sehen. Neben des Helligkeitsempfindens ermöglichen die 3 Zapfenarten die Wahrnehmung von Farben. Dabei haben die L-Zapfen bei 560 nm, die M-Zapfen bei 530 nm sowie die S-Zapfen bei 420 nm ihr Absorptionsmaximum [Lev+11].

Um die Wirkung eines physikalischen Farbreizes auf die physiologisch wahrgenommene Farbe zu beschreiben, wurde von der CIE 1931 das CIE-Normvalenzsystem eingeführt und basiert darauf, dass die 3 Zapfen auf Licht einer bestimmten Wellenlänge verschieden empfindlich reagieren. Die 3 Primärfarben entsprechen Rot, Grün und Blau, da diese die 3 Empfindlichkeitsmaxima aufweisen. Aus diesen Primärfarben lassen sich additiv Farben mischen.

Die Daten wurden dabei aus einem Experiment gewonnen, bei dem die Probanden drei Lichtquellen, welche Licht in etwa den höchsten Empfindlichkeiten der 3 Zapfenarten emittierten so regeln mussten, sodass eine wahrgenommene Farbe erzeugt wurde, welcher einer vorgegebenen monochromatische Referenzfarbe entsprach. Das Sichtfeld betrug dabei  $2^\circ$  (*Normbeobachter*). Daraus ließen sich die *color-matching functions* für die 3 Primärfarben bestimmen. Daraus ergibt sich der CIE-RGB Farbraum. Mithilfe dieser *color-matching*

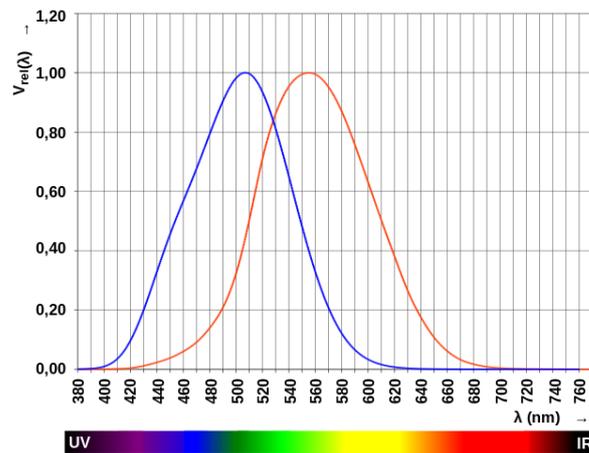


Abbildung 2.5.: Relative spektrale Hellempfindlichkeitsgrad  $V(\lambda)$  für das Tagsehen (rot) sowie  $V'(\lambda)$  für das Nachtsehen (blau). Entnommen aus [Hah10].

functions können somit vorhandene Strahlungsverteilungen  $P(\lambda)$  einer Lichtquelle unter Einbeziehung des und dem Maximalwert des fotometrischen Strahlungsäquivalents  $K_m$  in eine Farbvalenz überführt werden (siehe [RDH10]). In diesem Fall handelt es sich um den sogenannten *Tristimulus*, welcher den XYZ-Farbraum aufspannt [RDH10]. Dabei sind die Amplituden proportional zur Leuchtdichte [Poy03].

Um einen Farbort gut darstellen zu können, wurde eine Projektion aus dem aufgespannten XYZ-Raum in die Ebene  $xy$  entwickelt, die sogenannte CIE-Normfarbtafel. Dabei werden die Farbörter ohne Intensitätsbezug wiedergegeben. Die Koordinaten der *Normfarbtafel* werden wie folgt berechnet [Poy03]:

$$x = \frac{X}{X + Y + Z} \text{ sowie } y = \frac{Y}{X + Y + Z}$$

In Abbildung 2.21b auf Seite 28 ist solch eine Normfarbtafel abgebildet. Der XYZ-Raum hat jedoch den Nachteil, dass die Farbunterschiede hier nicht linear wahrgenommen werden. Zur Berücksichtigung von äquidistanten Farbabständen wurde der CIE-L\*a\*b\*- Farbraum entwickelt und kann in [Wil15; Poy03] näher studiert werden.

Neben der Farbe ist das Helligkeitsempfinden ebenfalls sehr wichtig und für diese Arbeit relevanter als die Farbwahrnehmung und wird im Folgenden eingeführt. In Dunkelheit werden durch die Stäbchen bereits bei einem Einfall von 50-100 Photonen Sinneseindrücke vermittelt [Ber17] welches einer 100-fachen Empfindlichkeit im Vergleich zu den Zapfen entspricht.

Bereits im Jahre 1834 stellte Ernst Heinrich Weber fest, dass bei geringen Reizintensitäten bereits eine geringere Erhöhung des Reizes ausreichte, um einen Unterschied (Just Noticeable Difference (JND, differentielle Wahrnehmbarkeitsschwelle)) zu Erkennen, als bei stärkeren Reizintensitäten. Die Schlussfolgerung war, dass der JND  $U_S$  vom Basisreiz  $S$  abhängt, wobei der Quotient konstant ( $k$ ) ist [Hop12]

$$U_S = k \cdot S \Leftrightarrow k = \frac{U_S}{S}$$

Fencher erkannte 1850, dass die JNDs von der Empfindungsintensitätsdifferenz  $dR$  bestimmt werden, sodass

mit dem differentiellen Basisreiz  $dS$  gilt [Hop12]

$$U_S = \frac{dS}{dR} \Leftrightarrow dS = \frac{dR}{U_S}$$

und es folgt

$$R = k^{-1} \cdot \ln(S) + C$$

wobei  $C = -\ln(S_1) \cdot k^{-1}$  der Integrationskonstante entspricht und der festzulegende Reiz  $S_1$  den Skalensprung bestimmt [Hop12].

Stevens stellte fest, dass das Weber-Fechner-Gesetz nur für kleine Abstände zwischen Reizstufen gilt und erweiterte dieses um die Relation zwischen der Reizstärke und der Empfindung in der sogenannten *Stevensche Potenzfunktion* im Jahr 1957 [Ste57]. Sei hier  $E$  die Empfindungsgröße, dann gilt für die Reizintensität  $S$  und Reizschwelle  $S_0$  folgende Potenzfunktion

$$E = k \cdot (S - S_0)^n \quad (2.12)$$

$k$  entspricht einer Skalierungskonstante und  $n$  dem vom Rezeptor abhängigen Konstante. Für das Helligkeitsempfinden gilt  $n \approx \frac{1}{3}$  [Ste57]. Somit ist für die Darstellung verschiedener (wahrgenommener) Helligkeiten auf einer Anzeige wie einem LCD oder OLED-Display, die Beachtung dieses Zusammenhangs notwendig. Dieser Zusammenhang wird bei Anzeigegeräten mithilfe sogenannter Opto-Electronic Transfer Functions (OETFs) und entsprechender inversen Electro-Optical Transfer Functions (EOTFs) realisiert. Klassischerweise wird bei Standard Dynamic Range (SDR)-Displays von *Gammakorrektur* oder kurz *Gamma* gesprochen. Dabei wird der Zusammenhang zwischen Bilddaten und der tatsächlichen Leuchtdichte durch eine Potenztransformation aus Formel (2.13) realisiert.

$$G : [0; 1] \rightarrow [0; 1], x \mapsto x^\gamma = G(x) \text{ mit } \gamma \in \mathbb{R} \quad (2.13)$$

Der Begriff *Gamma* bezieht sich dabei auf die EOTF und *Degamma* auf die OETF. Als Exponent für die obige Funktion wird oft der  $\gamma = 2,2$  gewählt [Poy03] und gilt insbesondere im Automobil. Manche Geräte, wie beispielsweise *Apple* Geräte können davon deutlich abweichen.

In Abbildung 2.6 wird die Anwendung von Gamma und Degamma innerhalb eines Bilddatenpfades dargestellt, auf dem auf physikalisch linearen Daten gearbeitet wird („Prozessing“). Der Eingangsdaten befinden sich in der wahrnehmungslinearen Domain und werden hier als  $R'G'B'$  bezeichnet. Nach der Gammaprozessierung werden diese in eine zur Leuchtdichte (physikalisch) proportionalen Domain umgewandelt und prozessiert (hier  $RGB_{lin}$ ). Schließlich werden die Daten wieder zurück-transformiert (Degamma) und können weiter an das Display übertragen werden. Das Display selbst setzt die Potenzfunktion mit dem entsprechendem  $\gamma$  um. Ein Cathode Ray Tube (CRT, Kathodenstrahlröhrenbildschirm) hat physikalisch bedingt inhärent eine Übertragungsfunktion mit einem  $\gamma = 2,5$  und ist ein sehr praktischer Zufall, dass die Funktion der menschlichen Wahrnehmung nahe kommt. Bei Displays, welche das Licht über Transistoren modulieren arbeiten in der optimalen Annahme linear. Die naheliegende Idee, die Bilddaten direkt physikalisch linear zu speichern und zu übertragen würde jedoch dazu führen, dass der Speicherbedarf und auch die Anforderung an die Übertragungsstrecke bezüglich der Datenrate massiv steigen würde. Als Beispiel wird eine Kontrastempfindlichkeit von 1% angenommen, welche zur Unterscheidbarkeit zwischen der Umgebungshelligkeit führt.



Abbildung 2.6.: Gamma Prozessierung im Datenpfad. Überführung zur physikalisch linearen Domain (*Gamma*) für das Processing und Umkehrung (*Degamma*). Entnommen aus [Sch13].

Dadurch wäre für ein Kontrastverhältnis von 100 : 1 mit  $\frac{100}{1\%} = 10000 \hat{=} 14 \text{ Bit}$  notwendig. Für die *Kodierung* mit der Gammafunktion werden für dieses Kontrastverhältnis rechnerisch 9 Bit benötigt, von denen nach [Poy03] 8 Bit ausreichen. Somit stellt die Speicherung und Übertragung von wahrnehmungslinaren Daten eine sinnvolle Reduktion der Datenmenge dar und wird zu diesem Zweck angewandt. Es sei jedoch angemerkt, dass für automotive Anwendung das Kontrastverhältnis um den Faktor 10 bis 12 höher sein muss, um eine adäquate Qualität und unterscheidbare Abstufungen zu liefern.

Da Displays „physikalisches“ Licht emittieren, ist es zweckmäßig, dass die in dieser Arbeit eingeführten Methoden und Algorithmen auf physikalisch linearen Daten basieren. Somit müssen die Gammakurven der genutzten Displays gemessen werden, um die Bilddaten entsprechend dem Gamma eines Displaymodells anzupassen. In Hardware wird die Gammafunktion basierend auf den Messwerten in Lookup-Tables (LUTs, Umsetzungstabelle) gespeichert und Taktweise ausgelesen. Die Verläufe der Kurven selbst werden typischerweise mit einem Spektrometer gemessen und können durch Fitting modelliert werden [Bla16a]. In diesem Fall mit der SpectraScan PR-740. In Abbildung 2.7 sind zu einem Prototyp die einzelnen Gammakurven zu den 3 Primärfarben sowie für die Kombination (grau nach weiß) aufgetragen. Die Messwerte sind absolut.

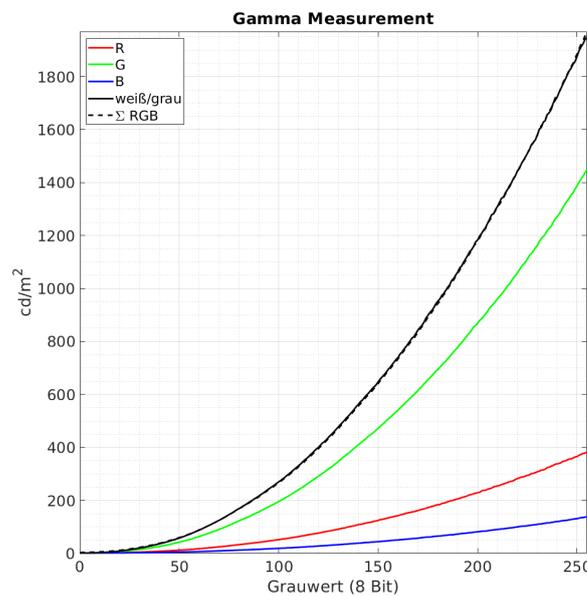


Abbildung 2.7.: Messergebnisse der durchgeführten Gamma-Messung (SpectraScan PR-740) für die 3 Primärfarben, sowie für weiß.

Es ist deutlich erkennbar, dass die grünen Subpixel den größten Anteil zur Gesamthelligkeit haben und die blauen den geringsten. Die Ursache liegt in der spektralen Hellempfindlichkeit  $V(\lambda)$  aus Abbildung 2.5 (Seite

17). Für die Integration in einen Algorithmus werden die Kurven jeweils auf ihre Maximalwerte normiert. Zur besseren Übersicht über einige wichtige Farbräume und ihre Zusammenhänge werden in Abbildung 2.8 abgebildet. Im unteren umrandeten Teil ist die Anwendung der Gammakorrektur, also der Potenzfunktion angedeutet, um von  $R'G'B'$  die Bilddaten in eine physikalisch proportionale Domain zu überführen. Für die Umkehrung wird die Degamma Funktion angewandt.

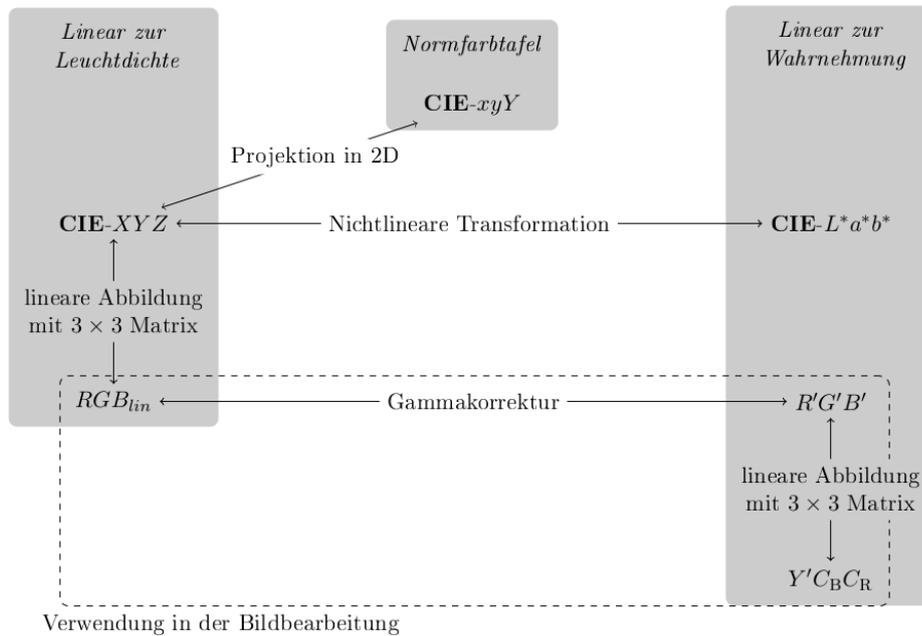


Abbildung 2.8.: Beziehungen und Zusammenhänge der Farbräume. Entnommen aus [Kar17] in Anlehnung an [Poy03].

### 2.3. Aktuell verbreitete Displays und deren Funktion

In diesem Abschnitt werden die grundlegenden Funktionsweisen von LCDs sowie OLED-Displays vorgestellt. Die beiden hier vorgestellten Displaytypen basieren auf einer matrixmäßigen Anordnung der Pixel. Dabei wird beim Treiberschema zwischen Passiv-Matrix und Aktiv-Matrix Ansteuerung unterschieden [BHJ16]. Passiv-Matrix wird jedoch nur noch bei geringen Auflösungen verwendet, da es mit der Länge der Leiterbahnen zu Spannungsabfällen durch den Widerstands- und Kapazitätsbelag kommt, welche wiederum kompensiert werden müssen. Zudem muss die Adressierungsfrequenz sehr hoch sein, da keine Kapazität vorhanden ist, welche die Spannung über einem Pixel aufrechterhalten könnte, um die Leuchtdauer zu verlängern.

Im Falle der Aktiv-Matrix-Ansteuerung, werden auf dem Panel sogenannte Thin-Film-Transistors (TFTs, Dünnschichttransistor) [Wei62] genutzt um einen Pixel anzusteuern. Charakteristisch ist hier das Vorhandensein einer Kapazität, welche für die Ansteuerung der OLED oder LC-Pixel während der nicht angesteuerten Phase genutzt wird. Der schematische Aufbau eines TFTs wird in Abbildung 2.9 abgebildet. Sein Funktionsprinzip entspricht dem eines Metal-Oxide-Semiconductor Field-Effect Transistor (MOSFET) (Funktionsweise siehe [Raz01, S. 9 ff.]). Der „Kanal“ wird oft mit amorphem Silizium ( $\alpha$ -Si) oder Low Temperature Poly Silicon (LTPS) realisiert [Bro16], während das Gate aus transparentem Indiumzinnoxid (ITO) besteht.

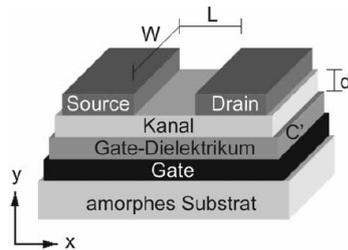


Abbildung 2.9.: Aufbau eines TFT. Entnommen aus [Gör08].

### 2.3.1. Aufbau und Funktion von LC-Displays

Die hier vorgestellten Dimming Methoden werden bei LCDs angewandt, deren Funktionsweise in diesem Abschnitt erläutert wird. Die Basis des LCD stellt der Flüssigkristall dar. Dieser Stoffe besitzen sowohl isotropische als auch anisotropische Eigenschaften eines Kristalls [Lue01]. Für die Displayanwendung müssen die Kristallmoleküle eine gewisse Vorzugsrichtung aufweisen. Die in der sogenannten *nematischen* Mesophase befindlichen Flüssigkristalle erfüllen diese Eigenschaft und lassen sich durch elektrische Felder beeinflussen. Dieser Effekt kann genutzt werden, um die Polarisationsrichtung des Lichts sinnvoll zu drehen. In Abbildung 2.10 wird beispielhaft die nematische Mesophase der Flüssigkristalle dargestellt. Eine Vorzugsrichtung ist erkennbar. Im Folgenden wird auf den Aufbau eines LCD eingegangen. Dazu wird zunächst die sogenannte

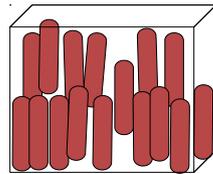


Abbildung 2.10.: Visualisierung der nematischen Mesophase des Flüssigkristalls. Entnommen aus [Sch13].

TN-Zelle (*twisted nematic*) im Querschnitt in Abbildung 2.11 betrachtet. Die einzelnen Komponenten sind farblich voneinander abgehoben. Die grundlegendste Komponente ist dabei der *Flüssigkristall*, welcher im E-Feld freien Zustand durch den *Orientation Layer* in eine helix-förmige Struktur gezwungen wird (Abbildung links). Diese beiden Layer bestehen aus Polyimiden und haben jeweils orthogonale Bürstungsrichtung um die Flüssigkristalle auszurichten.

Die *transparenten Elektroden* sind mit Indiumzinnoxid realisiert. Diese sind dazu da, um ein elektrisches Feld um den Flüssigkristall einer Zelle anzulegen und so die Helixform zu begradigen (Abbildung rechts). Die Flüssigkristallmoleküle richten sich parallel zum E-Feld aus. Bei Aktiv-Matrix-Ansteuerung werden sie durch TFTs gesteuert. Die *Polfilter* stehen orthogonal zueinander und lassen nur linear polarisiertes Licht in der entsprechenden Richtung passieren [You97]. Das zunächst diffuse, weiße Licht wird durch die Hintergrundbeleuchtung erzeugt und trifft auf den ersten Polfilter, welcher nur Licht mit linearer Polarisation durchlässt. Dieses Licht passiert den Flüssigkristall und wird (Abbildung links, spannungsfrei) in der Polarisationsrichtung orthogonal gedreht, sodass der zweite Polfilter (oder auch Analyzer) passiert werden kann. Sind die Flüssigkristallmoleküle jedoch wie rechts gezeigt, parallel zum E-Feld ausgerichtet, so wird die Polarisationsrichtung des Lichts nicht geändert und wird vom zweiten Polarisationsfilter absorbiert. Das Pixel bleibt dunkel.

Diese Art von Zelle wird auch als *normally-white* bezeichnet, da im spannungsfreien Zustand, das Licht die Zelle passieren kann. Daneben existieren auch *normally-black* Varianten.

Mithilfe der eingestellten Potenzialdifferenz lässt sich also die Transmission eines (Sub-) Pixels einstellen.

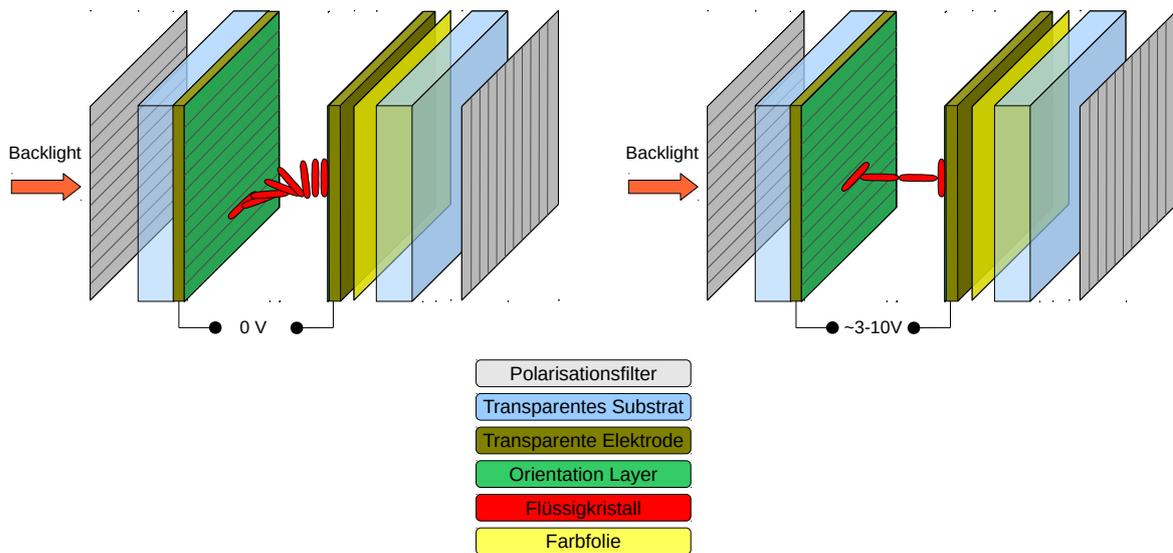


Abbildung 2.11.: Schematischer Aufbau einer TN-LC-Zelle. Entnommen aus [Sch13], angelehnt an [Lue01].

Um farbige Bilder zu erzeugen, wird ein Pixel in 3 Subpixel mit jeweils den Farbfiltern für Rot, Grün und Blau unterteilt. Diese Unterteilung basiert auf der additiven Farbmischung dieser Primärfarben, um einen wahrgenommenen Farbeindruck aus der Superposition der drei zu erzeugen. Mit diesen Primärfarben wird der Farbraum aufgespannt, welcher mit diesem Anzeigegerät wiedergegeben werden kann. Daneben existieren auch Pixelanordnungen mit einem zusätzlichen transparentem, also weißem Subpixel [Shi+18]. Neben der hier vorgestellten, schnellen TN-Zellen existieren weitere Lösungen, um bestimmte Eigenschaften zu verbessern. So punkten beispielsweise die In-Plane-Switching (IPS) Zellen [Aok06] mit einem stabileren Blickwinkel und Vertical Alignment (VA) mit einem hohen Kontrast. Im Automobil werden für hochwertige Displays in den meisten Fällen IPS Panels genutzt.

In Abbildung 2.12 ist der gesamte Aufbau, inklusive der Hintergrundbeleuchtung (Backlight) dargestellt. Die letztere wird auch als BLU bezeichnet. Es sind auch die beiden typischen Arten der BLU dargestellt. Links das sogenannte *Direct-Lit*, bei dem die Leuchtmittel, in diesem Fall LEDs direkt in einer Matrix-Anordnung hinter dem LC-Panel liegen, während rechts das *Edge-Lit* angedeutet ist. Hier sind die LEDs an mindestens einer der Seiten angeordnet (siehe auch Abbildung 2.13). Das Licht muss insbesondere bei Edge-Lit mit einer *Light-Guide-Plate* nach vorne, also in Richtung des Betrachters umgelenkt werden [Käl99; Käl12], um orthogonal zum LC-Panel die Flüssigkristallzellen zu durchqueren. In dieser Arbeit wird die Anzahl der LEDs bzw. von einzeln ansteuerbaren LED-Gruppen mit  $N_L$  bezeichnet. In Abschnitt 2.5 wird auf die Funktionsweise des Dimmings eingegangen wobei zwischen Edge- und Direct-Lit BLUs unterschieden wird. Neben dem Diffusor können weitere Folien zum optischen Stack der BLU hinzukommen, um die Effizienz zu erhöhen. Zu nennen ist hier der *Brightness Enhancement Film* sowie der Dual Brightness Enhancement Film von 3M™ (DBEF) [All+04; Web+96; Jon+99; 3M 08]. Die Lichtausbeute kann hier durch Lichtbündelung nach vorne bis zu 60% erhöht werden. Dies ist besonders wichtig, da die BLU den Hauptverbraucher eines LCDs darstellt. So kann der Energieumsatz der BLU etwa 90% betragen [HLL08]. Je nach Aufbau

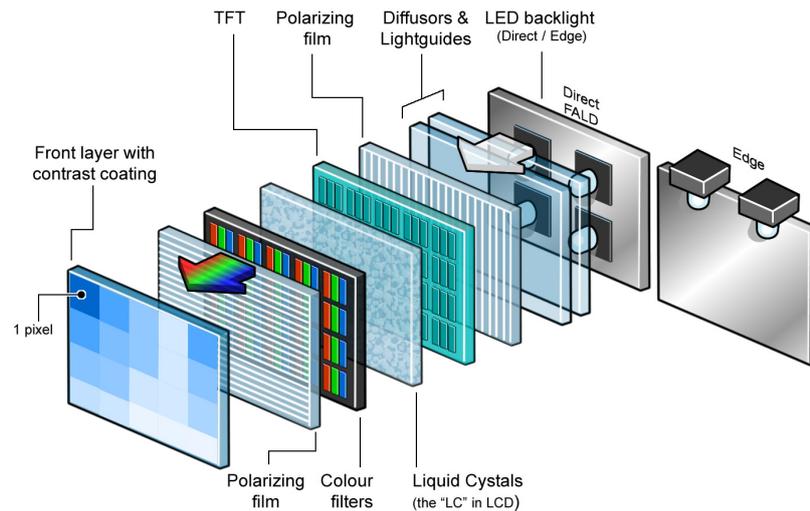


Abbildung 2.12.: Schematischer Aufbau eines LCDs mit Hintergrundbeleuchtung. Entnommen aus [Ras16].

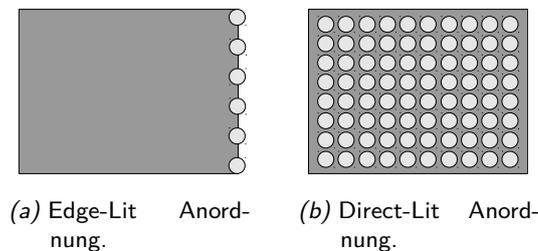


Abbildung 2.13.: Schematische Anordnung der LEDs in einer BLU. Entnommen aus [Sch13].

und Effizienz kommen noch 7 % als Lichtausbeute der BLU beim Betrachter an [Hd09]. Die Kavitäten in denen die LEDs sich befinden, können zudem mit Reflektoren und Linsen weiter optimiert werden, um die Lichtausbeute weiter zu steigern, wie beispielsweise in [Gu+20].

In Abbildung 2.14 ist die Relation zwischen optischer Distanz und Abstrahlwinkel einer LED in einer Direct-Lit BLU visualisiert. Insbesondere für den Automobileinsatz ist die Tiefe bzw. Dicke der BLU wesentlich für ihre Verwendung. Über den LED-Abstand (LED-Pitch) wird diese optische Distanz ebenfalls beeinflusst. Des Weiteren können monochromatische LEDs (blau) genutzt werden, um die Distanz zu verkleinern, da sich diese einer optimalen Punktquelle noch weiter annähern und sich die Lichtverteilung präziser kontrollieren lässt [Gu+20]. Die sogenannte *Color Conversion* (in Weiß) muss dann weiter hinten im optischen Stack mithilfe von Phosphor Dotierung realisiert werden. Die Ansteuerung der LC-Zellen erfolgt für in modernen Displays über die Active-Matrix-Ansteuerung mit TFTs. Das Konzept ist in Abbildung 2.15 abgebildet. Die Zeilen werden nacheinander adressiert, die Spalten für gewöhnlich parallel. Schaltet ein Transistor durch, so wird die parallel zum LC geschaltete Kapazität  $C$  vom Spaltentreiber mit einer analogen Spannung, die eine gewünschte Transmission einstellt, geladen und hält die Spannung bis zur nächsten Adressierung. Dadurch bleibt auch in der nicht angesteuerten Zeit die Transmission des Pixels eingestellt. Die Zeilentreiber werden auch Gate-Driver genannt und die Spaltentreiber Source-Driver. Standard im Automobilbereich sind 60 Frames per Second (FPS) bzw. 60 Hz.

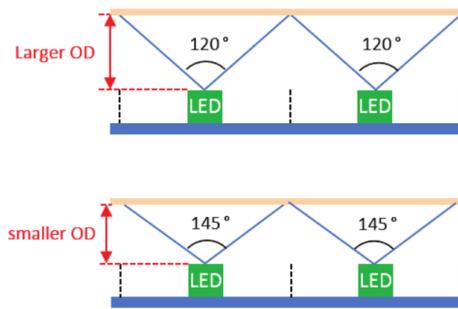


Abbildung 2.14.: Optische Distanz bei der Direct-Lit BLU. Entnommen aus [Xia+19].

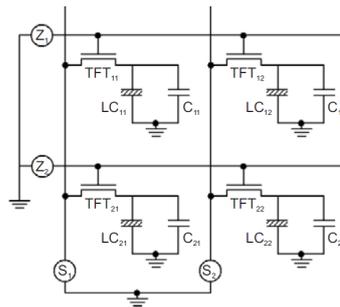


Abbildung 2.15.: Aktiv-Matrix Ansteuerung des LC-Panels. Entnommen aus [Gör08].

### 2.3.2. Grundlegender Aufbau von OLED-Displays

Anders als die LCDs benötigen OLED-Displays keine Hintergrundbeleuchtung. Eine OLED für sich betrachtet ist ein organischer Halbleiter, welcher durch Anlegen einer elektrischen Spannung sichtbares Licht emittiert (Elektrolumineszenz). In Abbildung 2.16 ist ein vereinfachtes Diagramm zur Veranschaulichung der Lichtemission dargestellt. Im organischen Halbleiter existiert das höchste besetzte Molekül-Orbital (*HOMO*, highest occupied molecular orbital) sowie das niedrigste unbesetzte Molekül-Orbital (*LUMO*, lowest unoccupied molecular orbital). Zwischen *HOMO* und *LUMO* ist eine Bandlücke vorhanden. Als Anode kommt hier oft transparentes ITO zum Einsatz. Durch Anlegen einer Potentialdifferenz an Kathode und Anode werden Elektronen und Löcher injiziert und durch das E-Feld transportiert. Dazu muss jedoch die Energiebarriere zwischen Kathode und *LUMO* sowie zwischen Anode und *HOMO* überwunden werden. Im sogenannten *Emission Layer* (EML) entstehen gebundene Elektronen-Loch-Paare (Exziton), welche bei der Rekombination Photonen einer bestimmten Wellenlänge emittieren [Jia18; Vol17]. Die Wellenlänge wird von der Bandlücke zwischen *HOMO* und *LUMO* bestimmt.

Moderne OLEDs bestehen aus mehreren Layern (Multi-layer OLED) zur Erhöhung der Rekombinations-Effizienz. Das Ziel ist die Angleichung der Energieniveaus, sodass eine Energiebarriere in mehrere kleinere aufgeteilt wird. In Abbildung 2.17 wird solch ein Aufbau dargestellt. Dazu werden der *Electron Injection Layer* (EIL) sowie der *Hole Injection Layer* (HIL) genutzt. Diese vereinfachen den Übergang von den Elektroden in das Device. Anders als bei anorganischen Halbleitern, können die Ladungsträger sich nicht frei bewegen, das sie an ihre entsprechenden Moleküle gebunden sind. Der Ladungstransport geschieht durch Reduktion und Oxidation benachbarter Moleküle und wird als *Hopping* bezeichnet [Rie14]. Um diesen Transport

### 2.3. Aktuell verbreitete Displays und deren Funktion

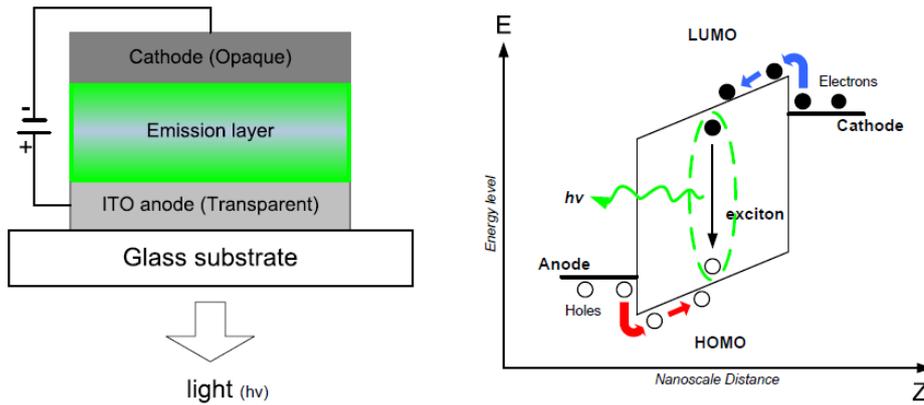


Abbildung 2.16.: Vereinfachtes Diagramm zur Funktionsweise der Lichtemission bei einem organischen Halbleiter. Entnommen aus [Jia18].

zu verbessern werden noch der *Electron Transport Layer* (ETL) sowie der *Hole Transport Layer* (HTL) eingeführt. Hiermit wird die Beweglichkeit gesteigert. Im *Emission Layer* (EML) entsteht das Quasiteilchen Exziton, wie bereits weiter oben beschrieben wurde und relaxiert dabei mit einer 25% Wahrscheinlichkeit unter Aussendung eines Photons [Vol17]. Für weiteres Studium der Funktionsweise von OLEDs wird auf [Tsu17; JX21] verwiesen. Diese OLEDs können nun in einer Matrix angeordnet als Pixel eines OLED-Panels

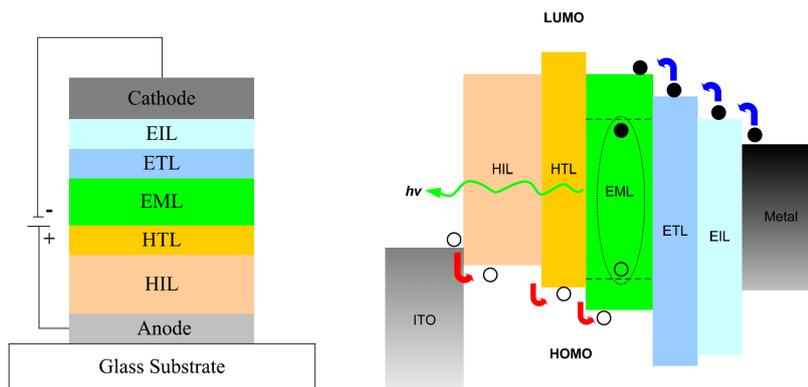


Abbildung 2.17.: Multilayer-Struktur einer OLED sowie schematische Energieniveaus. Entnommen aus [Jia18], original in [Tsu17].

gefertigt und als Display genutzt werden. Angesteuert werden die einzelnen selbst-emittierenden Pixel mit einer Active-Matrix-Ansteuerung (AMOLED), welche beispielhaft als 2T1C Realisierung in Abbildung 2.18 gezeigt wird. Hier sorgt die Kapazität  $C$  dafür, dass der Drive-Transistor  $TFT_2$  nach der Programmierzeit in Abhängigkeit von  $C$  leitet und somit ein Strom durch die OLED eingestellt werden kann.  $TFT_1$  wird durch die Zeilenadressierung durch  $V_{scan}$  durchgeschaltet und erlaubt das Aufladen von  $C$  während einer aktiven Zeile. Dabei wird  $C$  über  $V_{Data}$  auf den gewünschten Pegel geladen.

Ein großer Vorteil, welcher aus dem Aufbau eines OLED-Displays ergibt, ist die sehr hohe Flexibilität in Bezug auf Biegung des Panels. Dadurch wird so ein Display universeller einsetzbar. Ein weiterer wichtiger Vorteil ist der optimale Schwarzwert, da einzelne Pixel bei Stromlosigkeit auch kein Licht emittieren. Jedoch sind

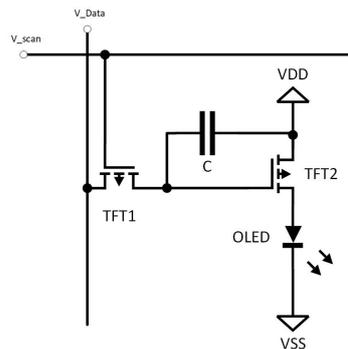


Abbildung 2.18.: OLED Aktiv-Matrix Ansteuerung mit der 2T1C Realisierung.

auch Probleme bekannt. So können keine sehr hohen Leuchtdichten erzeugt werden, da die OLED-Displays einen Alterungseffekt aufweisen, welcher der Abnahme der Effizienz der Pixel geschuldet ist. Die Pixel altern je nach Bildinhalt und Leuchtdauer verschieden was zu *Einbrenneffekten* führt [JVX15; Vol17]. Von Plasma-Display ist dieses Problem bereits bekannt. Diesen Nachteil teilen sich prinzipiell alle bisherigen selbst-emittierenden Displays. Am LME wurde bereits zur Erhöhung der Lebensdauer von OLED-Displays geforscht und Kompensationsmethoden- und Modelle erfolgreich entwickelt um den Nachteil für Consumer-Geräte reduzieren zu können [VX15; XD12; JX21].

Auf die Nutzung von OLED-Displays in der Automobilanwendung wird in Abschnitt 3.2 eingegangen.

Aufgrund von benötigten Schattenmasken für RGB-OLED Produktion, ist diese Technologie für mittlerer und größere Displays nicht wettbewerbsfähig zu LCD. Weitere Einblicke in LCDs und OLED-Displays werden in [Che+17a] gegeben. Im nächsten Abschnitt wird die Imperfektion eines LCDs aufgrund des sogenannten *Light-Leakage* erläutert.

## 2.4. Imperfektion von LCDs

Im Gegensatz zu OLED Displays (siehe Abschnitt 2.3.2), welche ein *perfektes* bzw. *optimales* Schwarz darstellen können, ist dies mit LCDs nicht möglich. Die OLED Pixel sind selbst-emittierend, sodass eine Abwesenheit des Pixelstromes dazu führt, dass keine Photonen emittiert werden. Das Pixel bleibt somit schwarz. Bei einem LCD wird das Licht wie in Abschnitt 2.3.1 beschrieben durch eine Lichtquelle erzeugt (BLU). Auch bei einem Grauwert von 0 wird das Licht nicht zu 100% blockiert, sodass der nicht blockierte Rest, ein Pixel grünlich-bläulich leuchten lässt. Dieses Artefakt, welches systeminhärent zu einem LC-Gerät gehört, wird als *Light-Leakage* bezeichnet [Kor+11; Che+17b; SWF12; Bur+13; Kim+09]. Die Ausprägung hängt von der Art der LC-Zelle ab. Der Leuchtdichtewert bei schwarzem Pixel wird *Blacklevel* oder *Schwarzwert* genannt.

Gerade in dunkler Umgebung führt das Light-Leakage zu einem störenden Artefakt, da schwarze Bereiche grünlich-bläulich erscheinen. Insbesondere in Automobilanwendung, wie in Abbildung 2.19 abgebildet, wirkt das Display deutlich vom Interieur abgesetzt und reduziert dadurch die wahrgenommene Wertigkeit. Insbesondere bei Premium-OEMs ist das Interesse jedoch groß, das Display möglichst integriert und *seamless*, also nahtlos ins Interieur-Design übergehen zu lassen. Durch mechanische Spannungen, welche im Fertigungsprozess entstehen, sowie kleine Abweichungen in den Zellen, kann das Light-Leakage *örtlich*

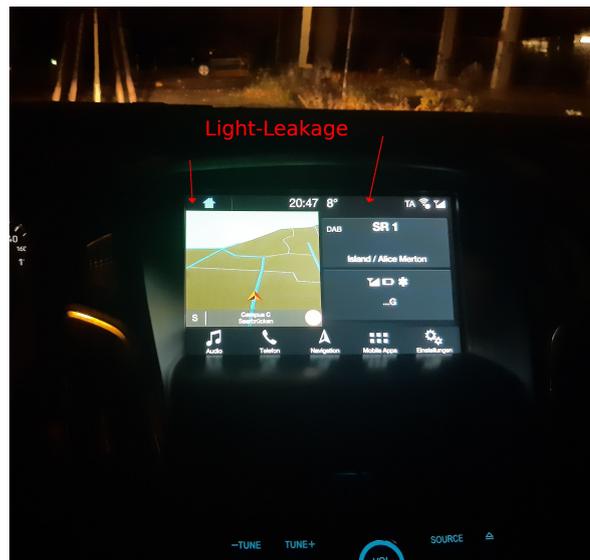


Abbildung 2.19.: Störendes Light-Leakage eines CIDs. Die markierten Flächen haben den Grauwert 0. Die Leuchtdichte ist jedoch auffällig störend  $> 0 \text{ cd m}^{-2}$ .

verschieden stark ausgeprägt sein. Diese Imperfektion wird als *Non-Uniformity* bezeichnet. Diese Non-Uniformity bei eingeschaltetem Backlight und schwarzem Bildinhalt, wird auch *Black-Mura* genannt [Bla18; LST09; Sys18]. Weitere Ursachen für solch eine Inhomogenität unabhängig von der LC-Zelle selbst, können auch Ungenauigkeiten der LED-Treiber (Stromeinstellung), sowie die Abweichung von LED-Kennlinien sein. In Abbildung 2.20 sind Messungen eines LCD Prototyps mit Direct-Lit BLU abgebildet. Der Blacklevel entspricht einem schwarzen Bildinhalt bei angeschaltetem Backlight und das All-On ist analog dazu ein weißes Bild. Die örtlichen Unterschiede sind hier hauptsächlich durch mechanische Spannungen am LC-Panel verursacht und resultiert in einem sehr stark schwankenden lokalen Kontrast. Dieser wird auch im Querschnitt dargestellt. Diese Imperfektion ist auch bei der Vermessung der Gamma-Kurve und des Gammuts sichtbar,

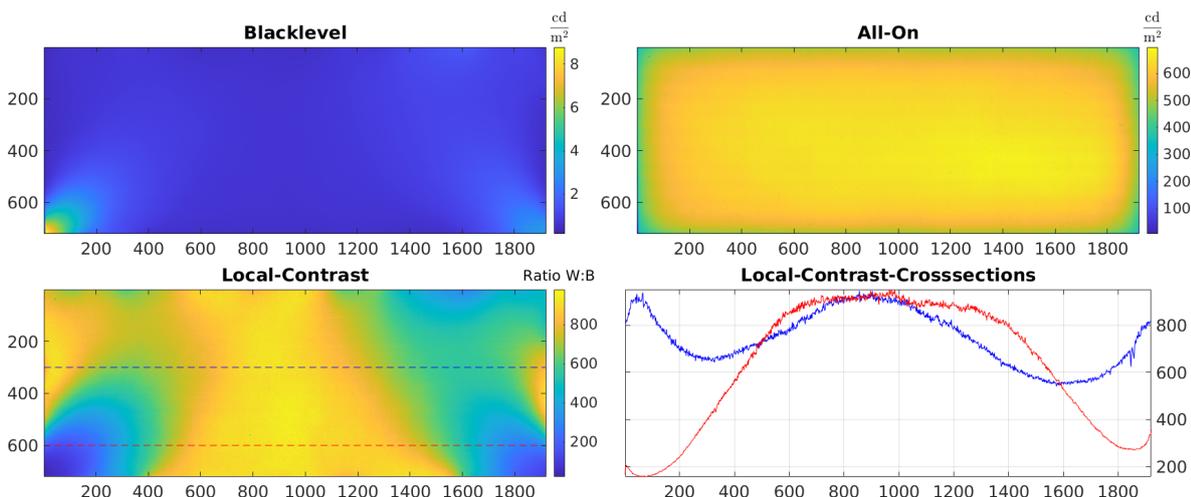


Abbildung 2.20.: Messung des Light-Leakage mit der Eldim UMaster und resultierender lokaler Kontrast.

wie deutlich aus Abbildung 2.21 hervorgeht. Gemessen wurde mit der SpectraScan PR-740 für alle Subpixel sowie für Weiß insgesamt. Hierbei handelt es sich um einen von oben verschiedenen Aufbau. In (a) ist die

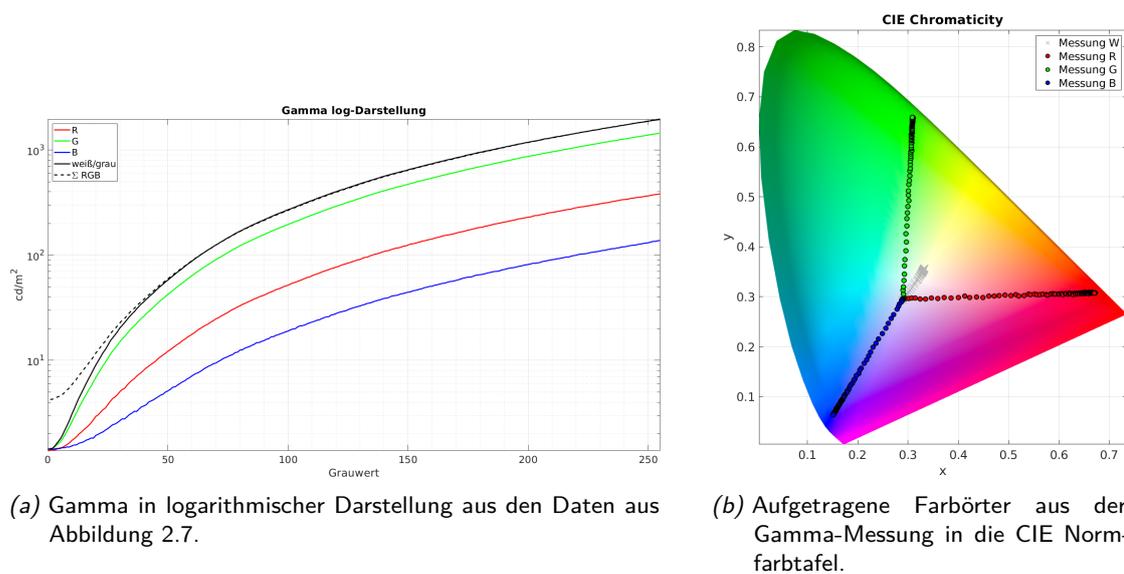


Abbildung 2.21.: Gamma-Messung und entsprechende Farbörter eines vermessenen LC-Panels. Die Imperfektion ist deutlich erkennbar.

Gamma-Kurve logarithmisch aufgetragen. Der Offset bei Grauwert 0 entspricht dem Schwarzwert. Werden die 3 Einzelmessungen aufsummiert (gestrichelt), so ergibt sich eine von der Weißmessung unterschiedliche Kurve. Das liegt daran, dass das Light-Leakage bei den Einzelmessungen gemessen wird, sodass sich in Summe der 3-fache Wert ergibt. In (b) sind die xy- Farbörter der einzelnen Messungen in der CIE-Normfarbtafel aufgetragen. Die Grauwerte bewegen sich von *innen* nach *außen*, wobei die äußersten Werte den Gammut des Displaysystems definieren. Gut erkennbar ist auch der Drift des Weißpunktes aus dem bläulichen Bereich hinaus. Es wird hier deutlich, dass dunkle Grauwerte zu einem weniger gesättigten Bild führen, da das Light-Leakage einen relativ großen Einfluss im unteren Bereich hat.

## Schlussfolgerung

Bei einem LCD ergeben sich folgende Imperfektionen und Nachteile gegenüber einem OLED Display

- Light-Leakage (schlechter Schwarzwert)
- Inhomogenitäten
- reduzierter Farbraum bei dunklen Inhalten
- Display Dicke fällt aufgrund der LCD Struktur deutlich größer aus

Trotz den Nachteilen, bietet die LCD Technologie jedoch auch Vorteile gegenüber OLED-Displays. Einer der wichtigsten ist die bekannte Lebensdauerproblematik [Vol17; Jia18; JVX15]. Daraus resultiert auch, dass bei einem OLED-Display die Spitzenleuchtdichte geringer ausfällt, da ansonsten die Lebensdauerproblematik weiter verstärkt wird. Zudem ist ein OLED-Modul für größere Diagonalen deutlich kostspieliger. Somit ist

ein LCD für die meisten Anwendungen im Automobil sinnvoller [Cum20b]. Auf die OLED-Anwendung im Automobil wird in Abschnitt 3.2 näher eingegangen.

Die Performance von LCDs kann mit Dimming Technologien weiter verbessert werden, um sich der Bildqualität eines OLED-Displays anzunähern. Insbesondere kann der Schwarzwert optimiert und die Stromaufnahme gesenkt werden. Dadurch wird auch der Farbraum bei dunklen Inhalten vergrößert [Sch+16a]. Somit wird das LCD auch für die Anwendung im E-Auto weiterhin und zukünftig relevant bleiben. Um die wahrgenommene Qualität zu steigern werden hoch-aufgelöste Direct-Lit BLUs benötigt (siehe Kapitel 4), wie im Verlauf der Arbeit näher erläutert wird.

## 2.5. Dimming Prinzipien

Dieses Kapitel dient der Einführung der grundlegenden Funktionsweisen des Global-Dimming und Local-Dimming. Dabei wird das Dimmingkonzept erläutert sowie der für das Local-Dimming wichtige Begriff der *Light-Spread-Function* (Abschnitt 2.5.2) eingeführt und die Pixelkompensation vorgestellt.

### 2.5.1. Konzept

In Abschnitt 2.3.1 wurde die Funktionsweise von LC-Displays eingeführt. Einen systeminhärenten Nachteil stellt hier das *Light-Leakage* dar, welches schwarze Flächen grau bzw. bläulich erscheinen lässt und die Bildqualität degradiert. Zudem ist die BLU der Hauptverbraucher beim LCD. Diese Gründe sprechen, wie auch schon in der Einleitung beschrieben, für die Nutzung eines Dimming Systems zur Reduktion des Strombedarfs, sowie zur Verbesserung der visuellen Qualität. Die Grundidee eines Dimming Systems wird in Abbildung 2.22 dargestellt. In (a) wird der Gesamtaufbau visualisiert und die Abhängigkeit des

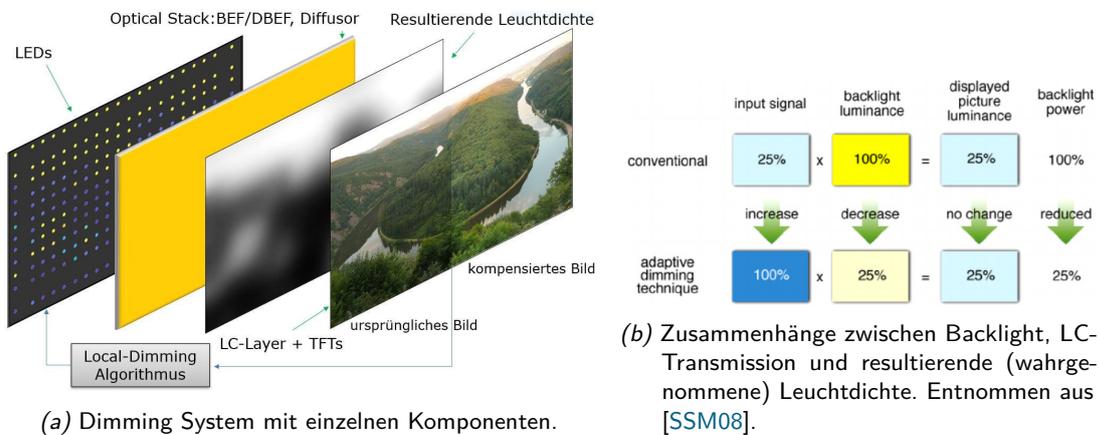


Abbildung 2.22.: Grundprinzip eines Dimming-Systems.

Backlights vom Bildinhalt über einen Dimming Algorithmus angedeutet. In (b) wird der Zusammenhang zwischen Backlight, der LC-Transmission und der resultierenden Leuchtdichte vereinfacht dargestellt. Die obere Reihe zeigt eine ungedimmte BLU bei einem Bildeingangswert von normierten 25%. Das LC-Panel moduliert die Hintergrundbeleuchtung durch einen Durchlass von genau diesen 25%. Dies resultiert in der wahrgenommenen Helligkeit beim Betrachter durch Multiplikation der beiden Intensitäten [Kim+09]. Hierbei beträgt jedoch der Strombedarf der BLU 100%. Beim Dimmen werden nun die Leuchtmittel in der BLU

verdunkelt und die Transmission der Flüssigkristall im LC-Panel im Gegenzug erhöht, sodass beim Betrachter dasselbe Helligkeitsempfinden hervorgerufen wird wie im konventionellen System. Dabei kann sich jedoch die Stromaufnahme der BLU erheblich reduzieren. Die Erhöhung der Transmission der Flüssigkristalle in den Pixels wird auch als *Kompensation* bezeichnet. Die BLU selbst kann dabei, wie in Abbildung 2.13 auf Seite 23 gezeigt aufgebaut sein. Grundsätzlich wird zwischen einer *Edge-Lit* und einer *Direct-Lit* BLU unterschieden.

Das Ziel eines Dimming Systems ist dabei die möglichst optimale Bestimmung der LED-Werte in Abhängigkeit der zugrundeliegenden BLU und des Bildinhaltes (siehe auch Abschnitt 2.5.3).

Aus Abschnitt 2.2 sind die Begriffe EOTF und OETF bereits bekannt. Insbesondere bei SDR Displays werden diese Funktionen durch eine Potenzfunktion (*Gammafunktion*) und deren Umkehrfunktion (*Degamma*) realisiert. Typischerweise handelt es sich bei dem Gammafunktion bei Automobilanwendungen um die Funktion aus Formel (2.13) von Seite 18 mit  $\gamma = 2,2$ .

In Abbildung 2.23 wird die Platzierung eines Dimming Algorithmus bzw. die Prozessierung der Bilddaten dargestellt. Nach der Gammatransformation stehen die linearisierten (proportional zur Leuchtdichte) Bilddaten zur Verfügung. Nach der Kompensation werden die Bilddaten wieder in die wahrnehmungslineare Domain durch eine *Degamma*-Funktion überführt. In Hardware werden Gamma sowie Degamma oft durch LUTs realisiert. Für die korrekte Prozessierung muss die Gammakurve gemessen werden. Typischerweise wird die Messung mit einem Spektrometer durchgeführt. In dieser Arbeit wurden diese Gamma-Messungen mit einer SpectraScan PR-740 [Pho] ausgeführt. Dabei werden die resultierenden Leuchtdichten der 3 Primärfarben vom kleinsten zum größten Grauwert auf dem Zieldisplay gemessen.

Die zum Dimming benötigten LEDs werden in der Regel mittels PWM betrieben. Dabei wird ein konstanter Diodenstrom eingestellt und die LEDs schließlich über den Duty-Cycle (*Pulsweite*) gesteuert [Sun12]. Ein großer Vorteil gegenüber einer Amplitudenmodulation (AM) Ansteuerung liegt in dem konstanten Farbort, da die Stromamplitude konstant bleibt.

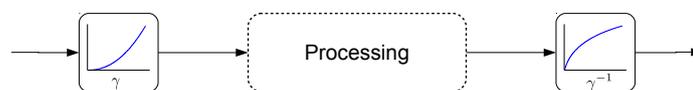


Abbildung 2.23.: Integration der eines Dimming Algorithmus zwischen der EOTF und OETF. Entnommen aus [Sch13].

In Abbildung 2.24 wird ein Überblick über die verschiedenen Dimming-Arten und ihren Beziehungen zur BLU dargestellt. In der mittleren Zeile befinden sich die Dimming Varianten, während oben die Backlight-Typen aus Abbildung 2.13 die Beziehungen verdeutlichen.

### 0D-Dimming

*0D-Dimming* wird auch als *Global-Dimming* bezeichnet. Dabei werden unabhängig von dem Aufbau der BLU, alle vorhandenen LEDs mit genau demselben Duty-Cycle (bzw. PWM-Wert) angesteuert und genau ein Wert berechnet. Diese Verfahren sind meist hardwarechonend zu realisieren, liefern jedoch keine hohen Einsparraten. Es wird jedoch nur ein einziger LED-Treiber benötigt.

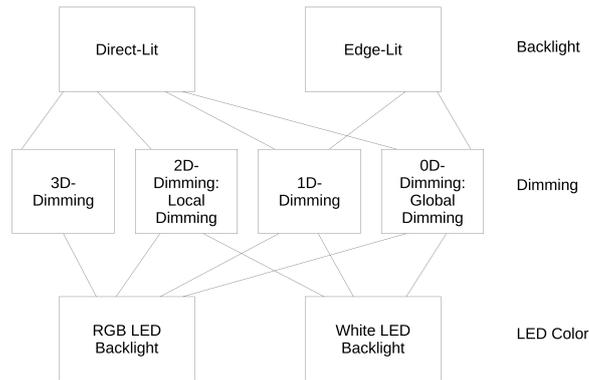


Abbildung 2.24.: Überblick der verschiedenen Dimming-Arten und ihren Beziehungen. Angelehnt an [XAJ12].

### 1D-Dimming

*1D-Dimming* bezeichnet das Dimmen von LEDs in horizontaler oder vertikaler Richtung. Dabei gibt bei Edge-Lit die LED-Anordnung vor, wie gedimmt werden kann. Aber auch Direct-Lit BLUs können 1D-gedimmt werden, wobei in diesem Fall mögliches Einsparpotential verloren geht. 1D-Dimming wird in dieser Arbeit schon zu Local-Dimming gezählt. Der erprobte SSC Local-Dimming Algorithmus wurde auf Edge-Lit Geräten validiert [SSX15; Jun+13; Grü+19; Sch+16a]. Dabei sind in der BLU meist  $N_L \in \{7, \dots, 24\}$  einzeln ansteuerbare LED-Stränge vorhanden.

### 2D-Dimming

*2D-Dimming* wird typischerweise mit Direct-Lit BLUs ermöglicht, da hier die LEDs in Gruppen oder einzeln, welche sich direkt hinter dem LC-Panel in einer Matrix-Anordnung befinden, gedimmt werden. Von dieser Art von Local-Dimming Algorithmen wird ein besonders hohes Einsparpotential erwartet. In dieser Arbeit werden neue Methoden für solch einen Algorithmus entwickelt und hardwarechonend in Hardware realisiert (Kapitel 4 bis 6). Beim SoA sind in der BLU  $N_L \in \{48, \dots, 240\}$  einzeln ansteuerbare LED-Stränge vorhanden. Der Trend ist dabei jedoch steigend und stellt neue Herausforderungen für Local-Dimming Systeme dar.

### 3D-Dimming

*3D-Dimming* basiert auf Direct-Lit BLUs, jedoch wird hier ein RGB-Backlight vorausgesetzt, sodass jede Farbe für sich selbst gedimmt werden kann. Dabei steigt jedoch auch der Hardwarebedarf für den Local-Dimming Algorithmus deutlich an, da jede LED-Wert-Verteilung für jede Farbe separat berechnet werden muss. Zudem hat sich das 3D-Dimming aus Kostengründen derzeit nicht durchsetzen können, sodass nicht näher darauf eingegangen wird.

In Abbildung 2.25 werden die einzelnen Dimming-Methoden anhand der LED-Wert-Verteilung verdeutlicht. Das 1D- und 2D-Dimming wird hier unter dem Begriff Local-Dimming zusammengefasst. Es wird nur zwischen Edge- und Direct-Lit BLUs unterschieden.

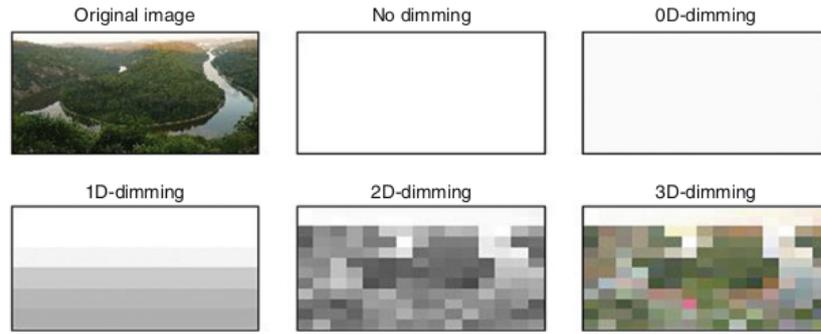


Abbildung 2.25.: Beispiel der verschiedenen Dimming-Methoden anhand der Saarschleife. Es werden die LED-Werte visualisiert. Je dunkler desto niedriger. Entnommen aus [XAJ12].

In Abschnitte 3.3 bis 3.4 wird der SoA des Global-Dimmings vorgestellt sowie weiterentwickelt und in Abschnitt 3.5.1 wird konkret auf den bekannten SSC Local-Dimming Algorithmus eingegangen, da er eine wichtige Basis für diese Arbeit darstellt. Um auf den am LME entwickelte SSC einzugehen, wird das Local-Dimming Modell vorgestellt. Es basiert auf den einzelnen Einflüssen der LEDs aus der BLU auf die Pixel benötigt, sodass sich das Resultat der Berechnung nahe am Optimum befindet. Darauf wird im nächsten Abschnitt eingegangen.

## 2.5.2. Light-Spread-Function und Einflussmatrix

In Abschnitt 2.3.1 wurde auf den Aufbau von LCDs eingegangen. Da die Pixel nicht selbst-emittierend sind, ist eine Hintergrundbeleuchtung notwendig, dessen Licht durch die einzelnen Sub-Pixel des LC-Panels moduliert werden. Somit wurde deutlich, dass eine einzelne LED unabhängig, ob bei Edge- oder Direct-Lit BLUs, deutlich mehr als ein Pixel beleuchten muss. Diese Lichtverteilung der einzelnen LEDs auf jeden einzelnen Pixel wird Light-Spread-Function (LSF) genannt und kann mit einem Leuchtdichtemessgerät flächig als Leuchtdichte gemessen werden [Sch+13; Sch+17; Che+09; Che+08]. Dazu wurde in dieser Arbeit das vorgestellte Kolorimeter Eldim UMaster verwendet. In Abbildung 2.26 werden zum Vergleich jeweils eine Messung einer Edge-Lit sowie einer Direct-Lit LSF gezeigt. Es wird deutlich, dass die Edge-Lit LSF einen deutlich *globalen* Charakter aufweist, während die Direct-Lit LSF *lokaler* wirkt. Weitere Edge-Lit LSFs sind in [Sch+13] zu finden. In Kapitel 4 wird genauer auf die Direct-Lit BLUs eingegangen. Diese LSF wird in manchen Algorithmen in sehr grober Form angenähert [LH10] und führt aufgrund von starker Abweichung zur Realität zu nicht optimalen Resultaten.

Das Überstrahlen einer LED in den gedachten Bereich einer anderen LED wird auch als *Crosstalk* bezeichnet [Li+07]. Dieser Crosstalk muss für die *optimale* Berechnung der LED-Werte Berücksichtigung finden. Dazu wird zunächst die Leuchtdichte einer LED  $l$  definiert als Skalarfeld

$$\left( L_{(i,j),l}^{\text{abs}} \right) =: \mathbf{L}_l^{\text{abs}} \in \mathbb{R}_{0,+}^{N_P^r \times N_P^c}, \quad \forall l \in \{0, \dots, N_L - 1\} \quad (2.14)$$

$N_L = \|\mathcal{L}\|$  ist dabei die Kardinalität der Menge der LEDs ( $\mathcal{L}$ ).  $N_P^c$  und  $N_P^r$  repräsentieren jeweils die Anzahl der Pixel Spalten bzw. Zeilen mit den entsprechenden Indizes  $(i, j)$ . Mit linearisiertem Pixelindex auch als  $L_{p,l}^{\text{abs}}$  abgekürzt und bezeichnet die Leuchtdichte emittiert von der LED  $l$  im Pixel  $p$ .

Die Messung bei maximaler LC-Transmission ( $t = 100\%$ ) und allen LEDs voll eingeschaltet wird als *All-On*

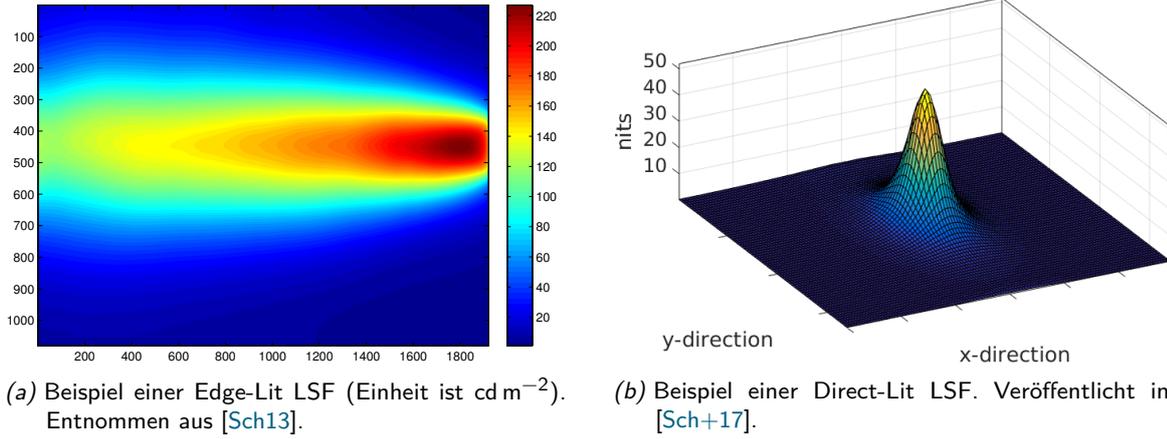


Abbildung 2.26.: Verschiedene Darstellungsmöglichkeiten von LSFs am Beispiel einer Edge-Lit LSF und einer Direct-Lit LSF.

bezeichnet und durch das Skalarfeld

$$\mathbf{L}_{\Sigma,t}^{\text{abs}} \in \mathbb{R}_{0,+}^{N_P^r \times N_P^c} \text{ mit } t = 1$$

repräsentiert. Für den später vorgestellten SSC Local-Dimming Algorithmus wird jedoch nicht die absolute Leuchtdichte, sondern der relative *Einfluss* einer LED  $l$  im Pixel  $(i, j)$  benötigt, um die LED-Werte berechnen zu können ( $\check{L}_l$ ). Diese Einflüsse werden durch Normierung wie folgt unter Zuhilfenahme von Formel (2.14) berechnet.

$$\check{L}_l := \left( \check{L}_{(i,j),l} \right) = \frac{L_{(i,j),l}^{\text{abs}}}{\sum_{l=0}^{N_L-1} L_{(i,j),l}^{\text{abs}}}, \text{ mit } \check{L}_l \in [0; 1]^{N_P^r \times N_P^c} \quad (2.15)$$

und impliziert

$$\sum_{l=0}^{N_L-1} \check{L}_l = \mathbb{1}^{N_P^r \times N_P^c} \quad (2.16)$$

Dieser Einfluss kann auch in einer sogenannten *Einflussmatrix* zusammengefasst werden wie in Formel (2.17) gezeigt. Dabei gilt  $N_P = \|\mathcal{P}\|$  und  $N_L = \|\mathcal{L}\|$ .

$$\left( \check{L}_{p,l} \right) =: \check{L} \in [0; 1]^{N_P \times N_L} \quad (2.17)$$

Im weiteren Verlauf der Arbeit werden die normierten Werte, wie die Einflüsse oder die darauf basierenden Berechnungen mit einem  $\check{\phantom{x}}$  dargestellt wie in Formeln (2.15) bis (2.17). Diese Einflüsse werden in der späteren Hardware benötigt und müssen in eine geeignete und effiziente Form gebracht werden, um eine Echtzeitprozessierung zu ermöglichen. Sie werden in der sogenannten *Pixelorder* gespeichert, welche in Abschnitt 3.5.1.2 vorgestellt wird.

### 2.5.3. Mathematische Beschreibung

Ein Local-Dimming Algorithmus kann mathematisch beschrieben werden wie beispielsweise in [Alb+10; SWF12]. Sei die Anzahl der LEDs bzw. die Anzahl der unabhängig ansteuerbaren LED-Stränge  $N_L$  und die Anzahl der Pixel  $N_P$ . Zunächst wird der Vektor  $\mathbf{v} \in [0; 1]^{N_L \times 1}$  definiert, welcher für alle linear indizierten LEDs den berechneten Duty-Cycle oder auch den PWM-Wert (normiert) beinhaltet [Ana08; Sun12]. Mit Formel (2.15) berechnet sich das normierte Backlight hinter einem Pixel  $(i, j)$  unter Berücksichtigung des Crosstalks zu dem Skalarfeld  $\check{\mathbf{B}}$ , welches auch als virtuelles Backlight bezeichnet wird. Die Berechnungsvorschrift ist in Formel (2.18) dargestellt.

$$\check{\mathbf{B}} := (\check{b}_{i,j}) = \sum_{l=0}^{N_L-1} \check{L}_{(i,j),l} \cdot \mathbf{v}(l) \quad (2.18)$$

mit  $\check{L}_l := (\check{L}_{(i,j),l})$

Seien die Post-Gamma Pixelwerte an linearisierter Position  $p$  definiert als Vektor  $\mathbf{r} \in [0; 1]^{N_P \times 1}$ , dann ist das Ziel eines Local-Dimming Algorithmus an jeder Pixel-Position die Zielleuchtdichte mindestens zu erhalten, da die Pixel-Transmission nur kleiner oder gleich 100 % eingestellt werden kann. Mit der Einflussmatrix aus Formel (2.17) muss also gelten:

$$\check{\mathbf{L}} \cdot \mathbf{v} \geq \mathbf{r} \quad (2.19)$$

Die Leistungsaufnahme  $P$  einer LEDs ist proportional zu ihrem PWM Wert bzw. Duty-Cycle und somit auch für die BLU betrachtet zur Summe aller Duty-Cycles [Bur+13; AKX08].

$$P \propto \sum_{l=0}^{N_L-1} \mathbf{v}(l) \quad (2.20)$$

Damit lässt sich die Reduktion der Leistungsaufnahme, welche durch die Verringerung des Strombedarfs realisiert wird, also das *Powersaving* (PS) als Mittelwert über alle LEDs wie folgt berechnen:

$$PS = 100\% - \frac{\sum_{l=0}^{N_L-1} \mathbf{v}(l)}{N_L} \quad (2.21)$$

Dabei sei  $N_L$  die Anzahl der LEDs bzw. LED-Stränge in der BLU. Somit ergibt sich als Ziel eines Dimming-Algorithmus die Reduktion des Powersavings bei gleichzeitiger Sicherstellung der notwendigen Leuchtdichte hinter den Pixeln mit dem Index  $p$ . Hieraus und mit Formeln (2.17) und (2.19) lässt sich ein lineares Optimierungsproblem formulieren wie in Formel (2.22) beschrieben wird.

$$\min \left\{ \sum_{i=0}^{N_L-1} \mathbf{v}(i) : \sum_{l=0}^{N_L-1} \check{L}_{p,l} \cdot \mathbf{v}(l) \geq \mathbf{r}(p), \forall p \in \{0, \dots, N_P - 1\} \text{ mit } \mathbf{v} \in [0; 1]^{N_L \times 1} \right\} \quad (2.22)$$

Um die Komplexität zu verdeutlichen wird ein modernes Display mit einer Auflösung von  $720 \times 1920$  angenommen, somit müssen die Ungleichungen aus Formel (2.22) für  $720 \cdot 1920 \cdot 3 = 4147200$  Subpixel erfüllt und eine Lösung pro Frame in Echtzeit berechnet werden. Bei einer FPS von 60 Hz liegt das zeitliche

Limit also bei  $< 16,6\bar{6}$  ms. Der am LME entwickelte SSC Algorithmus soll dieses Problem lösen wie später in Abschnitt 3.5.1 für Edge-Lit BLUs erläutert wird.

#### 2.5.4. Pixelkompensation

In Abbildung 2.22b wurde deutlich, dass die Transmission der LC-Zellen erhöht werden muss, wenn das Backlight hinter diesen Zellen gedimmt wird, um die äquivalente resultierende Leuchtdichte zu erhalten. Anderenfalls führt eine Verringerung der LED-Werte sowohl bei Global-Dimming als auch bei Local-Dimming zu einem Abdunkeln des Gesamtbildes. Aus diesem Grund muss die Transmission, welche über die Grauwerte eingestellt werden, erhöht werden.

In manchen Dimming Systemem, wird die Pixelkompensation wie beispielsweise in [Zha+19] mithilfe einer Logarithmusfunktion durchgeführt, anstelle wie in Abbildung 2.23 gezeigt, unter Berücksichtigung der tatsächlichen Degamma-Funktion. Dies führt jedoch zwar zu einer Art *Tonemapping* und einem höheren Kontrast durch die Graustufen, jedoch wird damit auch der Bildinhalt manipuliert und ist nicht im Sinne des Informationserhaltes. Insbesondere im Automobil muss das angezeigte Bild jedoch dem angedachten Bild entsprechen und darf sich nicht hinsichtlich der Farbe nach dem Dimmen ändern. Aus diesem Grund wird die Pixelkompensation vor der Anwendung der Degamma-Funktion ausgeführt [KD06; SJX12; Sch17; Jun14; Che+06; Kor+11] und kann mit Kenntnis des Backlights  $\check{b}_p$  hinter dem Pixel  $(i, j)$  mit dem Wert  $r_p$  durchgeführt werden. Dabei sind alle Werte auf den Bereich  $[0; 1]$  normiert.

$$r_p^{\text{comp}} = \frac{r_p}{\check{b}_p} \quad (2.23)$$

$r_p^{\text{comp}}$  entspricht dabei dem kompensierten Wert. In Abbildung 2.27 ist die Funktionsweise anhand eines Testbildes verdeutlicht. Im linken Bild ist ein Backlight zu sehen, welches auf der linken Seite komplett an (ohne Dimming) ist, während es auf der rechten Seite gedimmt wurde. Im rechten Bild sind die entsprechenden Transmissionswerte aus Formel (2.23) dargestellt. Die adaptierten Werte durch die Pixelkompensation erscheinen hier heller, da die Grauwerte höher sind. Werden diese jedoch mit dem gedimmten Backlight des Displays überlagert, so stellt sich die ursprüngliche Leuchtdichte ein. Dabei wird jedoch die Leistungsaufnahme de BLU reduziert.

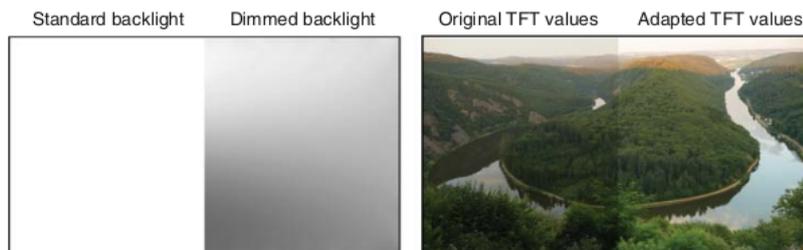


Abbildung 2.27.: Beispiel der Pixelkompensation anhand der *Saarschleife*. Entnommen aus [XAJ12].

## 2.6. Hardwareumsetzung eines Bildverarbeitungsalgorithmus

### Repräsentation von Bildern

Um Bilder beispielsweise in Simulationen in Matlab nutzen zu können, wird zunächst auf deren Repräsentation eingegangen.

Nach der Bildakquisition mithilfe von CCDs oder Complementary metal-oxide-semiconductor (CMOS) Sensoren werden die zum Photoneneinfall proportionalen Werte durch eine Gamma-Funktion in Grauwerte überführt und als Subpixel-Werte gespeichert. Für eine detaillierte Beschreibung der Bildakquisition, wird auf [Poy03; Gon08] verwiesen. Dort wird auch auf die Orts- als auch auf die Wert-Diskretisierung eingegangen. Typischerweise werden Bilddaten mit einer Auflösung von  $b = 8$  Bit pro Farbkanal gespeichert [Poy03].

Im weiteren Verlauf der Arbeit wird die Anzahl der Pixelspalten mit  $N_P^c$  und die der Zeilen mit  $N_P^r$  bezeichnet. Die Gesamtanzahl ergibt sich dann zu  $N_P = N_P^r \times N_P^c$ . Die Menge aller Pixel wird mit  $\mathcal{P}$  beschrieben, deren Elemente die Pixel  $p_{i,j}$  sind. Der Index  $i$  beschreibt die Zeilen- und  $j$  die Spaltennummer. Der Index  $k$  repräsentiert die Primärfarben R, G und B in gerade dieser Reihenfolge.

Mathematisch lässt sich ein digitalisiertes Bild somit als Vektorfeld

$$I : \{0, \dots, N_P^r - 1\} \times \{0, \dots, N_P^c - 1\} \subset \mathbb{N}_0 \times \mathbb{N}_0 \rightarrow \{0, \dots, 2^b - 1\}^3 \subset \mathbb{N}_0 \times \mathbb{N}_0 \times \mathbb{N}_0 \quad (2.24)$$

beschreiben. Jedem Koordinatentupel wird ein Werte-Tripel zugeordnet. Dabei ist  $b$  die Anzahl der Bits welche für die quantisierten Werte zur Verfügung stehen. Typischerweise gilt für die Pixel-Werte  $I(i, j, k) \in \{0, \dots, 255\}$  mit  $b = 8$  für den wahrnehmungslinaren Bereich.

Ferner wird der Mittelwert als Average Pixel Level (APL) bezeichnet und ist für definiert als

$$\text{APL}(I) = \frac{1}{3 \cdot N_P} \cdot \sum_{i=0}^{N_P^r-1} \sum_{j=0}^{N_P^c-1} I(i, j, 1) + I(i, j, 2) + I(i, j, 3) \quad (2.25)$$

Für die physikalisch lineare Domain wird dieser Mittelwert als Postgamma APL beschrieben. Oft wird jedoch statt auf den Subpixel auf daraus resultierenden Maximalwerten pro Pixel gearbeitet, da die RGB-Subpixel denselben Lichtbeitrag von der (weißen) Hintergrundbeleuchtung erfahren. In der Anwendung in Abschnitt 3.5.1.1 wird darauf zurückgegriffen. Um diese Daten von einer *Quelle* zu einer *Senke* übertragen wird die Topologie eines LCD-Moduls und der Pixelpipeline bzw. des *Bilddatenstromes* vorgestellt.

### Topologischer Aufbau eines LCD-Moduls

Wird von einem *Display* gesprochen, so ist ein *System* bestehend aus Einzelkomponenten gemeint, welches neben der BLU und des LC-Panels auch Schnittstellen und Treiber beinhaltet. Dabei spielt der sogenannte TCON eine wichtige Rolle, da dieser IC die Steuerung der Zeilen- und Spaltentreibern der TFT-Matrix übernimmt, welche typischerweise direkt auf dem TFT-Panel integriert (Chip on Glass) sind. Zudem stellt der TCON die letzte Instanz dar, welche noch Manipulationen am Bilddatenstrom durchführen kann. Somit kann dieser als eigentliche *Bildsenke* betrachtet werden. Mit einer enthaltenen Gamma-LUT werden Grauwerte zugehörigen TFT-Spannungen zugeordnet. Ein LED-Controller zur Steuerung der LED-Treiber ist entweder als dedizierter IC ausgeführt oder auch im TCON selbst integriert. Die entsprechenden Backlight-Parameter werden von einem übergeordneten System definiert. In Abbildung 2.28 ist der Aufbau eines typischen LCD-Systems abgebildet.

Die Bilddaten selbst werden von der *Bildquelle* (e.g. Graphics Processing Unit (GPU, Grafikprozessor)) erzeugt und müssen zum TCON übertragen werden. Dabei wird unterschieden, ob die Daten *innerhalb* eines Systems (auch Board-to-Board oder Chip-to-Chip) oder über *externe* Schnittstellen übertragen werden (insbesondere *Consumer* Bereich). Davon abhängig existieren verschiedene (physikalische) Übertragungstechniken und Protokolle. Der TCON hat typischerweise eines der *embedded* Schnittstellen in Hardware implementiert. In der Abbildung sind diese in der Farbe des TCONs in hellem Orange visualisiert. An dieser Stelle sei angemerkt,

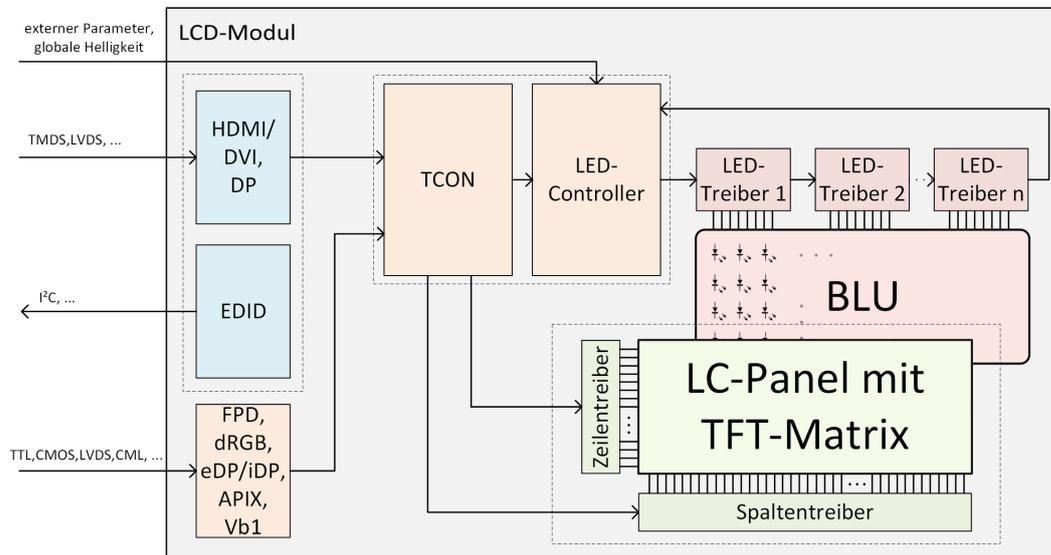


Abbildung 2.28.: Aufbau und Zusammensetzung eines LCD-Moduls.

dass zwischen dem physikalischen Layer und dem Protokoll bei der Datenübertragung unterschieden werden muss. Im Bereich der Displayelektronik wird im allgemeinen Sprachgebrauch und auch in wissenschaftlichen Publikation der Begriff Low Voltage Differential Signaling (LVDS) fälschlicherweise für den Flat Panel Display Link (FPD) mit definiertem Bitmapping [Sta05; Tex20] genutzt. Tatsächlich beschreibt LVDS nur einen physikalischen Layer und findet bei mehreren Protokollen Anwendung. Das LVDS-Interface wird von ANSI (American National Standards Institute) sowie TIA/EIA (Telecommunications Industry Association/Electronic Industries Association) definiert [EIA01]. Unter anderem wird FPD bei den Prototypen im Rahmen dieser Arbeit genutzt. Daneben wird oft High Definition Multimedia Interface (HDMI), welches elektrisch auf Transition-Minimized Differential Signaling (TMDS) basiert, verwendet.

### Pixelpipeline

Der Begriff der *Pixelpipeline* (Bilddatenstrom) beschreibt im Wesentlichen das sequentielle *Schieben* von Bilddaten (e.g. Grauwerte) mit dem Takt der *Pixelclock*. Dabei werden die Daten durch mehrere Register geschoben. Darunter wird beispielsweise das sequentielle Durchlaufen der Bilddaten von der Bildquelle (*Source*) zur Bildsenke (*Sink*) verstanden. Die Senke ist dabei oft der TCON eines Displays (vgl. Abbildung 2.28). Im Folgenden ist  $f_p$  als Frequenz der Pixelclock definiert. Zum besseren Verständnis ist in Abbildung 2.29 eine Pixelpipe dargestellt. Der Bilddatenstrom kann bei Bedarf auch aufgespalten (e.g. parallele Pixelpipes mit halber Frequenz ( $\frac{f_p}{2}$ )) werden, um die Übertragungsfrequenz zu reduzieren. Dies ist insbesondere bei sehr hohen Auflösungen auf diverser Hardware, wie beispielsweise bei FPGAs

sinnvoll, da strukturbedingt (Signallaufzeit) die *kritischen Pfade* sehr kurz werden und somit die Tiefe der kombinatorischen Logik deutlich reduziert werden muss. In dieser Arbeit werden die später vorgestellten Algorithmen basierend auf der Pixelpipeline im FPGA implementiert. Um eine definierte Übertragung zu gewährleisten, muss das sogenannte *Interface Timing* beachtet werden, welches als nächstes vorgestellt wird.

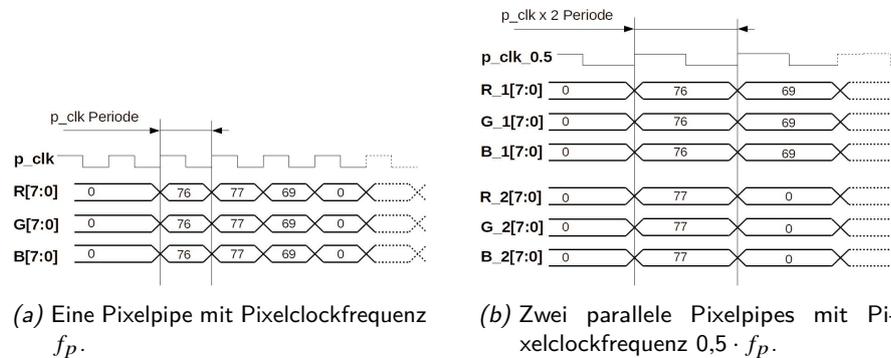


Abbildung 2.29.: Pixelpipeline mit Beispielwerten sowie Aufteilung in zwei parallele Pipes.

## Interface Timing

Der Begriff *Interface Timing* beschreibt hier die zeitlich wohldefinierten Signale und zugehörige Bilddaten, welche vom TCON zu einem bestimmten Zeitpunkt erwartet werden. Historisch bedingt, existieren aus der Zeit der Röhrenbildschirme sogenannte *Blanking-Zeiten*, welche dazu genutzt wurden den Elektronenstrahl auf die nächste Zeile (horizontales Blanking, H-Sync) oder auch von der letzten zur ersten Zeile (vertikales Blanking, V-Sync) zu lenken. Diese Zeiten wurden durch die Sync-Signale (H- und V-Sync) markiert. Die aktive Zeit der validen Bilddaten wird mit *DE* (Data Enabled) angezeigt. In Abbildung 2.30 wird die relative Lage des Bilddatenstromes zu den Sync-Signalen deutlich. Standard Timings werden durch die Video Electronics Standards Association (VESA) definiert [Vid13]. Die wichtigsten Parameter wie *Pixelclock*, *Displayauflösung* und auch die *Blanking-Zeiten* werden als Extended Display Identification Data (EDID) in einem Electrically Erasable Programmable Read-only Memory (EEPROM, elektrisch löschbarer programmierbarer Nur-Lese-Speicher) gespeichert und werden über einen Inter-Integrated Circuit (I2C, I-Quadrat-C) Bus an die Bildquelle übertragen. Die Definition der einzelnen Abschnitte in einer EDID können in [Vid06] nachgelesen werden. Im Rahmen dieser Arbeit mussten auch EDID Daten erzeugt werden, um der Bildquelle ein definiertes Timing aufzuzwingen. Dieses Vorgehen ist insbesondere sinnvoll, wenn nicht sichergestellt werden kann, dass die Bildquelle manuell auf ein definiertes Timing eingestellt werden kann.

Der Begriff *Timing* umfasst auch die Übertragung über verschiedene Schnittstellen und Standards. In Abbildung A.1 ist das Schema des auf LVDS basierenden FPD Link dargestellt. Insbesondere sind das Pinmapping und die relative Lage zur übertragenen Clock relevant. Wichtig ist hierbei, dass die Frequenz der einzelnen Bits bei diesem Interface der 7-fache Frequenz der Übertragungsclock entspricht (bei 2 parallelen Kanälen also genau dem 3,5-fachen der Pixelclock). Die Übertragung von Bilddaten ist bei OLED- und LC-Displays identisch. Bei LCDs kommt jedoch zusätzlich die notwendige Ansteuerung der Hintergrundbeleuchtung hinzu.

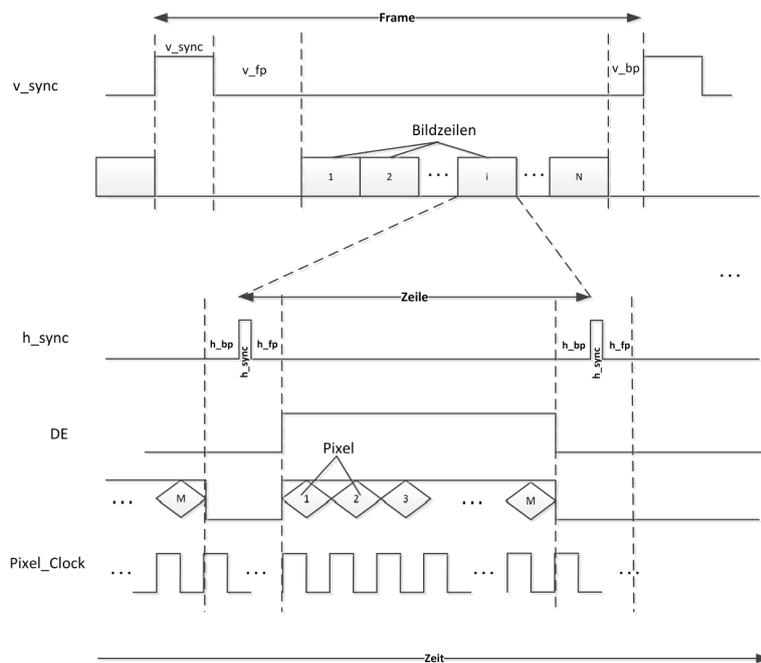


Abbildung 2.30.: Timing Diagramm eines Bilddatenstromes. Entnommen aus [Jun14].

### Funktionsweise von FPGAs

Die in dieser Arbeit entwickelten Algorithmen wurden mithilfe von FPGAs in Hardware realisiert. Diese stellen einen IC dar, welcher in seiner Funktion durch die vorhandene Struktur konfigurierbar ist. Mit einem FPGA kann eine beschriebene Hardware in einem versatil einsetzbaren IC umgesetzt werden. In dieser Arbeit wird das FPGA in die Pixelpipeline eingesetzt. Auf dem Bilddatenstrom wird die Pixelkompensation ausgeführt und ein modifizierter Bilddatenstrom ausgeliefert. Außerdem werden die Ansteuerwerte für die LEDs bestimmt und an die LED-Treiber gesendet. In Abbildung 2.31 ist die grundlegende Struktur eines FPGAs dargestellt. Die sogenannten Configurable Logic Blocks (CLBs, konfigurierbarer Logik Block) stellen die konfigurierbaren Komponenten dar [Xilb]. Diese sind über eine programmierbare *Switch Matrix (Programmable Interconnect)* an weitere Komponenten angebunden. Für das Speichern von Daten direkt auf dem FPGA sind SRAM Blöcke vorhanden, welche ebenfalls angesteuert werden können. Für die Kommunikation zur Außenwelt dienen feste I/O Blöcke, welche auch Serializer und Deserializer (SerDes) besitzen und werden beispielsweise die Serialisierung der Bilddaten benötigt (e.g. für LVDS). Die Slices innerhalb der CLBs bestehen aus LUTs sowie Flip-Flops (FFs, bistabile Kippstufe) und dienen der Umsetzung der synchroner und sequentieller Logik sowie reiner Kombinatorik. Der detaillierte Aufbau eines Slices der 7er FPGA Serie von Xilinx ist in [Xilb] zu finden. Das in dieser Arbeit genutzte Xilinx Kintex 7K325T besteht aus 50950 Slices, welche insgesamt 20800 6-Eingang-LUTs und 407600 FFs beinhalten [Xil18].

Um die gewünschte Hardware bzw. deren Funktion zu modellieren, wird eine sogenannte Hardware Description Language (HDL, Hardwarebeschreibungssprache) verwendet. Sie ermöglicht die *textuelle Beschreibung* von Hardware auf verschiedener Abstraktionsebene. Es existieren zwei sehr verbreitete HDLs, nämlich VHDL (Very High Speed Integrated Circuit Hardware Description Language) sowie Verilog. Die Erzeugung der *Konfiguration* für ein reales FPGA besteht aus mehreren Schritten, welche in Abbildung A.2 als Ablauf des *Hardware Design Prozesses* abgebildet sind. Dabei wird aus der HDL Beschreibung eine *Netlist* erzeugt,

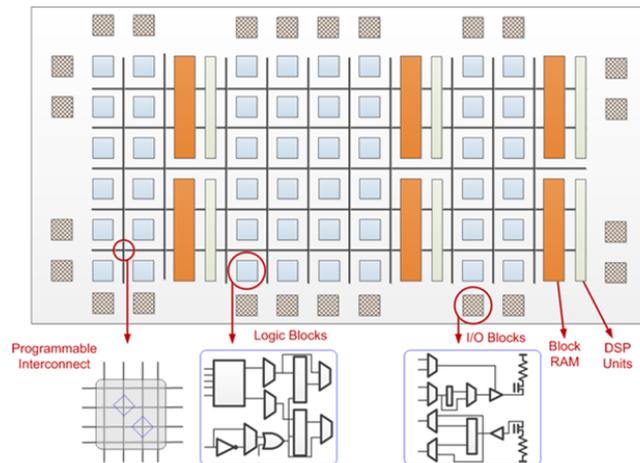


Abbildung 2.31.: Grundlegende Struktur und Komponenten eines FPGAs. Entnommen aus [Xila].

welche danach auf das Ziel-FPGA *gemapped* und *geroutet* wird. Für weitere Einblicke wird auf [Bai11] verwiesen.

Im Vergleich zum ASIC bieten FPGAs den Vorteil, dass die gesamten Entwicklungskosten gering sind und eine kürzere Entwicklungszeit benötigt wird. Zudem lässt sich die Konfiguration im Feld ändern. Dadurch ist die Time-to-Market kürzer. Gleichzeitig ist ein FPGA bei hohen Stückzahlen teurer als ein ASIC und hat einen größeren Leistungsumsatz, da mehr Komponenten versorgt werden müssen [Ges14]. Die maximal mögliche Clock-Frequenzen sind technologisch bedingt ebenfalls geringer als in ASICs.

Die für diese Arbeit relevanten Grundlagen und Konzepte wurden bis hier eingeführt, sodass im nächsten Kapitel auf die Anforderungen an Displays im Fahrzeug eingegangen werden kann. Daneben wird der Einsatz von OLED Displays diskutiert und auch ein Global-Dimming Algorithmus vorgestellt. Zudem wird auch auf den SoA des SSC Local-Dimmings besprochen, welcher die Ausgangslage für die in Kapitel 5 vorgestellten Methoden und Konzepte darstellt.

# 3. Besonderheiten und Lösungsansätze in Automobil-Anwendungen

In Kapitel 1 wurde bereits kurz auf die besonderen Herausforderungen für Displays im Automobil eingegangen und auch die Wichtigkeit eines qualitativ hochwertigen Displays begründet. Für den Fahrzeugführer soll insbesondere das Instrument Cluster Display (ICD, Kombi-Instrument Anzeige) eine hohe Wertigkeit aufweisen und bestimmten Anforderungen genügen. Für die Beifahrer und insbesondere zukünftig dank des Fortschritts im Bereich des autonomen Fahrens, werden weitere Displays wie die Center Information Displays (CIDs, Zentralanzeige) für das Entertaining sowie Bedienung des Fahrzeugs immer wichtiger.

Dieses Kapitel stellt die An- sowie Herausforderungen dieser Displays im Automobilbereich vor und geht auf die Problematik von OLED Displays im Automobil ein. Es wird zudem ein verbesserter Global-Dimming Algorithmus (Abschnitt 3.4) vorgestellt, welcher eine Vorstufe zu Local-Dimming darstellt, sowie die Funktion und SoA des SSC Local-Dimming Algorithmus (Abschnitt 3.5.1) eingeführt. Diese Methoden dienen der Reduktion des Strombedarfs, sowie der inhärenten Verbesserungen der Bildqualität.

## 3.1. Anforderungen an Displays im Fahrzeug

Aufgrund der speziellen Anwendung, insbesondere im Bereich der ICDs sind im Automobil Bilder mit speziellen informativen Elementen wie Tachometer, Drehzahlanzeigen und sicherheitsrelevante Warnsymbole üblich, da das Display als Schnittstelle zwischen Mensch und Auto fungiert. Diese Schnittstelle und der entsprechende Bildinhalt wird als Human-Machine-Interface (HMI, Mensch-Maschine-Schnittstelle) bezeichnet. Da diese Displays die klassischen analogen Kombi-Instrumente zunehmend ersetzen, sind die Anforderungen an sie sehr hoch [Bla18]. Mindestanforderungen zur Lesbarkeit (z.B. Helligkeit der Symbolgrößen) und Bewertungsmethoden sind zudem spezifiziert [DIN17b].

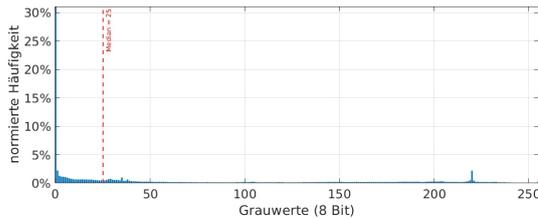
In Abbildung 3.1 ist ein typischer HMI-Inhalt und ein natürliches Bild abgebildet. Darunter sind die zugehörigen, normierten Histogramme. Es ist zu Erkennen, dass das HMI Bild sehr künstlich ist, dadurch wird aber auch ein hoher Kontrast geschaffen. An den Histogrammen werden weitere Unterschiede deutlich. Bei dem HMI Bild befinden sich 50% der 8Bit Grauwerte unter dem Wert 25 während bei der Saarschleife diese Schwelle bei 80 liegt. Des Weiteren ist ersichtlich, dass über 30% der Pixel in a) den Wert 0 haben, also Schwarz dargestellt werden sollen. Darüber hinaus ist ein kleiner Peak im helleren Bereich sichtbar, welcher für den Kontrast zu den dunklen bzw. schwarzen Flächen sorgt. Bei einem natürlichen Bild ist die Verteilung der Grauwerte deutlich ausgewogener. Fast jeder Grauwert ist vertreten, die Übergänge sind *weicher* und somit auch niederfrequenter. Des Weiteren haben automotive HMIs häufig kreisförmige Strukturen, wie Abbildung 3.1a zeigt. Dies hat auch Einfluss auf Local-Dimming, wie sich noch zeigen wird. Die Hintergründe sind meist mit *uniformen dunklen Flächen* realisiert und ergeben sich aufgrund der besseren Lesbarkeit wie in Abbildung 3.2 aus [DIN17b] gezeigt. Abgebildet sind Vorschläge zur Kombination von Symbol- zu Hintergrundfarben, um die Lesbarkeit zu verbessern. Die letzte Zeile ("Schwarz") zeigt, dass ein



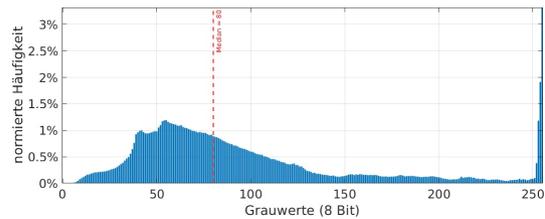
(a) Ein typisches HMI Bild.



(b) Die Saarschleife als Repräsentant eines natürlichen Bildes.



(c) Histogramm zu a) mit 256 Bins.



(d) Histogramm zu b) mit 256 Bins.

Abbildung 3.1.: Vergleich eines HMI mit einem natürlichen Bild mit zugehörigen Histogrammen. Diese sind auf die Anzahl der Pixel normiert.

*schwarzer Hintergrund* die meisten und höchsten Empfehlungen haben. Zudem wirkt das Display durch die großen schwarzen Flächen und kleineren hellen Inhalten *brillant*. Diese Brillanz wird durch ein Zusammenspiel vieler lokaler Kontraste empfunden [Bar16].

Da im Automobil die Anzahl der Displays zunimmt, nimmt der Wunsch nach *Senkung des Strombedarfs* in Folge der CO<sub>2</sub>-Emissionen einen sehr hohen Stellenwert ein. In Kapitel 1 wurde bereits auf die weiter ansteigende Pixel-Auflösung hingewiesen. Dies führt bei LCDs zu einem deutlich höheren Strombedarf, da eine Erhöhung der Pixeldichte zu einem geringeren Aperture Ratio (Öffnungsverhältnis) führt, wird zur Erzeugung derselben Leuchtdichte ein deutlich helleres Backlight benötigt. Die gestiegene Verlustleistung führt zu einem anspruchsvolleren *thermal Design* und *thermal Management* des gesamten Displaymoduls. Da die BLU bis zu 90 % zum Gesamtleistungsbedarf eines Displaymoduls beiträgt [HLL08], ist das die Entwicklung und Nutzung von Dimming-Methoden sehr sinnvoll. Zudem soll der *Kontrast* weiter steigen, was ebenfalls zu einer weiteren Erhöhung der Backlightelligkeit führt. Daneben muss ein großer *Temperaturbereich* abgedeckt werden und die Leuchtdichten müssen zwischen Tag und Nachtbetrieb mit einer geeigneten Quantisierung eingestellt werden können. Ein Display wirkt zudem hochwertiger, je geringer das sichtbare Light-Leakage des Panels ist. Dadurch lässt sich so ein Modul optimal in das Interieur-Design integrieren. Die bei diesem Punkt unschlagbaren OLEDs Displays haben jedoch bei hohen Leuchtdichten die Tendenz schneller zu Altern [JX21] und können nur einen begrenzten Strom also eine begrenzte Photonenanzahl und somit Leuchtdichte liefern.

Unabhängig von der Technologie muss die *Verlustleistung* in Form von Wärme abgeführt und zur Verbesserung der CO<sub>2</sub> Bilanz minimiert werden. Diese gilt für jede Komponente in einem Kraftfahrzeug. Die Lebensdauer, hochwertige Wahrnehmung und Reduktion des Strombedarfs sind jedoch entscheidende Herausforderungen. Meist wird eine Zielleuchtdichte vorgegeben und der Betrachter hat eine gewisse Erwartungshaltung zur Displayqualität, insbesondere bei *Premium* OEMs. Anders als im Consumer Bereich, indem ein Smartphone

### 3.1. Anforderungen an Displays im Fahrzeug

Tabelle B.1 — Kombinationen der Symbol-/Hintergrundfarbe

Hintergrundfarbe	Symbolfarbe						
	Weiß	Gelb	Orange	Rot <sup>a</sup> , Lila	Grün, Zyan	Blau <sup>a</sup> , Violett	Schwarz
Weiß	■	–	o	+	+	++	++
Gelb	–	■	–	o	o	+	++
Orange	o	–	■	–	–	o	+
Rot <sup>a</sup> , Lila	+	o	–	■	–	–	+
Grün, Zyan	+	o	–	–	■	–	+
Blau <sup>a</sup> , Violett	++	+	o	–	–	■	–
Schwarz	++	++	+	+	+	–	■

++ Bevorzugt  
 + Empfohlen  
 o Akzeptierbar bei großen Unterschieden der Farbsättigung  
 – Nicht empfohlen

<sup>a</sup> Reines Rot und Blau sollte vermieden werden, da die Augen aufgrund chromatischer Aberration beim Fokussieren auf diese Farben Probleme haben könnten

Abbildung 3.2.: Kombinationen der Symbol-/Hintergrundfarbe nach DIN EN ISO 15008:2017-07. Entnommen aus [DIN17b].

inklusive Display nach 2,75 Jahren [Dan21] ersetzt wird, wird von einem Auto eine vielfach höhere Lebensdauer erwartet [KZ19]. Neben den genannten Aspekten spielt der *Preis* also die Kosten eine sehr wichtige Rolle und ist entscheidend für die OEMs. Verbesserungen an der Qualität wie Schwarzwert oder Ablesbarkeit hat direkten Einfluss auf den Gesamtpreis, welcher wiederum Einfluss auf die Spezifikation der wahrnehmungsrelevanten Parameter (Qualität) ausübt. Diese Einflüsse bzw. Abhängigkeiten (*Magic Circle*) werden in Abbildung 3.3 dargestellt.

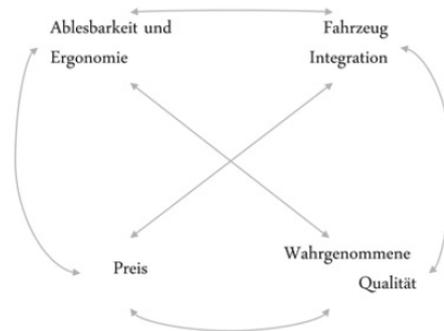


Abbildung 3.3.: Magic Circle der Zusammenhänge der Optimierung des Displayeigenschaften. Entnommen aus [Bau13].

Um den beschriebenen Anforderungen zu genügen, bedarf es weiterer Forschung und Verbesserung der einzelnen Komponenten wie bei Alterung von OLED Displays oder der Schwarzwert eines LCDs. Letzterer kann durch Dimming Algorithmen stark verbessert werden, wie im Laufe der Arbeit noch gezeigt wird. Aufgrund der Anforderungen und der OLED als Konkurrenz geht der Trend bei den LCDs von Edge-Lit BLUs auf Direct-Lit BLUs über und soll einen sehr hohen Kontrast und Powersaving ermöglichen. Dies ist unter anderem Thema von Kapitel 4 bis 6.

Die signifikanten Kosten eines Displaymoduls mit Direct-Lit BLU setzen sich zusammen aus den Kosten für

- LC-Panel (Glass)
- LEDs
- LED-Treiber
- Folienstack wie DBEF, Polarizer
- Hardware wie DC/DC Converter und TCON
- Entwicklung von Algorithmen zur Qualitätsverbesserungen

Letztere können beispielsweise Local-Dimming oder Global-Dimming IPs sein oder auch Algorithmen zur Verbesserung der Ablesbarkeit bei Sonneneinstrahlung. Implementiert werden diese letztendlich im Datenpfad des Bilddatenstromes nach Möglichkeit direkt im TCON. In Abschnitt 3.3 bzw. Abschnitt 3.4 wird ein sehr hardware-schonende und kostengünstige Methode vorgestellt, welcher zwar in seiner Performance begrenzt ist, aber in bestimmten Fällen angewendet werden kann.

Um Kosten zu senken und die Expertise von verschiedenen Zulieferern und OEMs im Displaybereich bündeln zu können und gemeinsame Spezifikationen festzusetzen wurde eine Automotive-Gruppe im Rahmen des Deutschen Flachdisplay-Forum e.V. (DFF) aufgebaut. Auch die *Universität des Saarlandes* ist DFF Mitglied. Dort werden zukünftige Trends diskutiert welche in den OEM Display Spezifikationen münden. In dieser Gruppe (German Automotive OEM Working Group) sind *VM, Audi, BMW, Daimler* und auch *Porsche* vertreten und sind seit 2008 Teil des DFFs. Sie liefert die *Display Specification for Automotive Application* [Ger18]. Spezifikationen zu Messungen (*Display Measurement Specification (DMS) for Automotive-TFT LCDs*) werden von der DFF entwickelt [Ger]. Eine Übersicht ist in Abbildung 3.4 dargestellt.

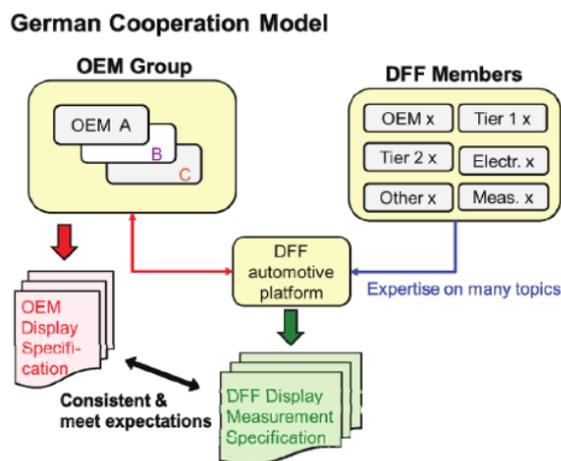


Abbildung 3.4.: Deutsches Kooperationsmodell mit DFF. Entnommen aus [Bla+16].

## 3.2. OLED Displays in der Automobilanwendung

OLED Displays sind aus dem Consumer-Bereich nicht mehr wegzudenken. Viele Smartphones und TV-Geräte setzen auf OLED-Displays. Ein bekannter Vorteil ist der perfekte Schwarzwert dieses selbst-emittierenden Displaytyps. Ein weiterer Vorteil ist die dünnere Bauweise durch die Abwesenheit einer BLU und sehr gute Formbarkeit und flexibler Krümmung. Im Automobilbereich sind diese Displays jedoch nur selten vorzufinden. Beispielsweise beim Audi e-Tron mit dem *Virtuellem Außenspiegel* [Aud] oder auch optional bei der Mercedes S-Klasse und auch in der elektrischen E-Klasse EQE für das Jahr 2022 geplant [OLE21]. In Abbildung 3.5 ist als Beispiel ein automotives OLED Displays der Firma BOE dargestellt. Dieses Display liefert jedoch eine typische Leuchtdichte von  $600 \text{ cd m}^{-2}$  und ein Maximum von  $803 \text{ cd m}^{-2}$  [Don+19]. Diese Werte sind jedoch bei hoher Sonneneinstrahlung nicht mehr ausreichend und somit nicht für exponierte Positionen im Automobil geeignet. Eine wichtige Ursache für die niedrige Leuchtdichte ist die *Alterung* des Panels in Abhängigkeit der Leuchtzeit und Stromdichte pro Pixel. Sie soll gerade bei langen Laufzeiten wie im Automobil, begrenzt werden. Ein Beispiel des *Einbrennens* ist in Abbildung 3.6 gezeigt. Es sind deutliche Symbole bei einem uniformen Bild zu erkennen. Diese werden aufgrund der Degradation der Stromeffizienz sichtbar.

Auch die Gründung der *OLED Working Group* im DFF im Jahr 2020 deutet an, dass im Automobilbereich ein Bedarf nach „echtem“ Schwarz noch nicht mit OLED-Displays erfüllt worden ist, da die Lebensdauerproblematik weiterhin besteht und die Displays im Verhältnis zu LCDs preislich nicht mithalten können. Mit der Arbeit dieser Gruppe an *OLED Specification for Displays in Automotive Application* sollen die automotiven Anforderung an OLED Displays weiter konkretisiert werden. Dem Problem der Lebensdauer wird versucht



Abbildung 3.5.: Ein 12,3 Zoll automotives OLED-Display von BOE. Entnommen aus [Don+19].

mit Kompensationsmethoden entgegenzuwirken. Dies ist auch Gegenstand der Forschung am LME [Vol17; JX21].

In Abbildung 3.7 wird ein Benchmark aus [Wan+17] zwischen einem OLED-TV und einem LC-TV mit Local-Dimming dargestellt. Dabei wird eine weiße Box auf schwarzem Grund kontinuierlich in seiner Fläche erhöht und ist auf der Abszisse als Anteil der Gesamtpixel aufgetragen. Die Ordinate zeigt die resultierende Leuchtdichte gemessen in dieser Box. An dieser Stelle soll nur der rote Plot betrachtet werden. Der Local-Dimming Teil wird später in Abschnitt 3.5 nochmal aufgegriffen. Deutlich erkennbar ist hier das Einsetzen einer *Derating*-Methode, welche die Stromdichte pro Pixel limitiert, wenn der Gesamtstrom einen



Abbildung 3.6.: Aufnahme eines *eingebrennten* OLED Displays. Gut erkennbar sind insbesondere die oberen Symbole durch eine verringerte Leuchtdichte. Entnommen aus [Vol17].

bestimmten Wert überschreitet. In diesem Fall also liefert das OLED-Display bei einem Weißanteil von  $\approx 10\%$  nicht mehr die Peakleuchtdichte. Bei einem weißen Bild liegt hier nur noch  $\approx 25\%$  der Peakleuchtdichte vor. Eine Derating Methode für den Automobileinsatz wird in [Wei16] vorgestellt. Es wird bei 2200 Tagesbetriebsstunden die Verringerung der Leuchtdichte von  $600 \text{ cd m}^{-2}$  um  $286,66 \text{ cd m}^{-2}$  aufgrund der Degradierung beschrieben. Mit einer Derating-Methode wird die Degradierung etwas gemindert, jedoch auf Kosten der mittleren Leuchtdichte über die Zeit.

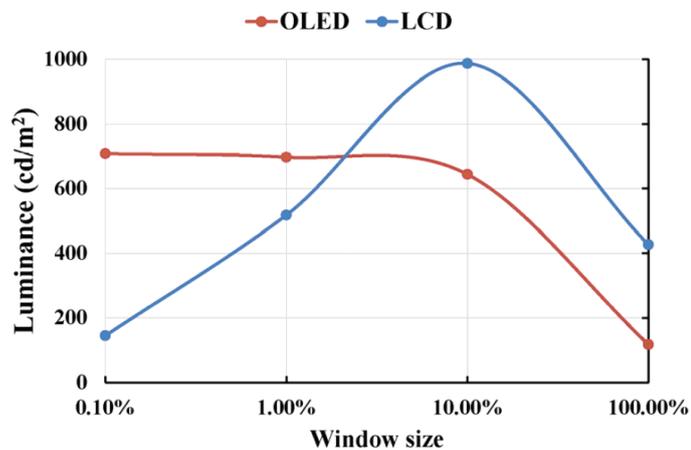


Abbildung 3.7.: Benchmark zwischen einem OLED TV und einem LC TV mit Local-Dimming. Entnommen aus [Wan+17].

Diese Nachteile wie

- niedrige Leuchtdichte
- Derating
- Temperaturempfindlichkeit
- sowie die resultierende Lebensdauerproblematik (Alterung)

sind ursächlichen Kostentreiber und motivieren folglich auch die weitere Optimierung von den schon sehr ausgereiften LCDs. Dadurch wird deren Qualität weiter gesteigert und neue Methoden zur Verbesserung entwickelt. Beispielsweise hoch-auflösenden BLUs und verbesserte Local-Dimming Algorithmen.

Insgesamt sind OLED-Displays jedoch unschlagbar beim *Darstellen* des Schwarzwertes und stellen einen Benchmark für LCDs dar. Es ist das Ziel mit LCD ein näherungsweise vergleichbares Bild erzeugen zu können. Dafür sind die neuen hochauflösenden BLUs (Kapitel 4) und neue Local-Dimming Algorithmen (Kapitel 5) sehr gut geeignet. Zudem ist OLED bei Bildern mit vielen dunklen Anteilen im Bereich des Powersavings bevorteilt, da die Pixel selbst das Licht emittieren und somit bildabhängig den Strombedarf reduzieren (unabhängig vom Derating betrachtet). Somit können OLED Displays auch als Powersaving Benchmark herangezogen werden.

### 3.3. Global-Dimming zur Reduktion des Strombedarfs

Eine kostengünstige und aus dem Consumer Bereich bekannte Methode ist das sogenannte Global-Dimming [Bho12; SSM08; KK09; HLL08], dessen Grundprinzip bereits aus Abschnitt 2.5 bekannt ist. Global-Dimming wird auch als Content Adaptive Backlight Control (CABC) oder 0D-Dimming bezeichnet, da alle vorhandenen LEDs mit einem einzigen Treiber angesteuert werden. Dadurch ist nur eine *globale* Einstellung der Hintergrundbeleuchtung möglich und erlaubt das Einsparen von Treibern, welche ansonsten für das Treiben separater LEDs benötigt würden. Die Grundidee ist, dass in einem Bild der maximale Grauwert nicht immer bei 100% liegt. Dadurch wird das Dimming erst ermöglicht. Beinhaltet das Bild jedoch den Maximalwert, so ist ein rechnerisch artefaktfreies Dimmen nicht mehr möglich. Um jedoch auch in diesem Fall den Strombedarf senken zu können, wird ein Verlust von Bildinformation in Kauf genommen. Um die Hardwarekosten für den Global-Dimming Algorithmus gering zu halten, basieren viele Methoden auf den Histogrammen der Bilder [IFP05; HLL08]. Dabei werden als unwichtig erachtete Bilddaten *verworfen* und das Histogramm gestreckt. Dabei wird auch der höchste vorhandene Grauwert reduziert. In Abbildung 3.8 wird das Prinzip dargestellt.

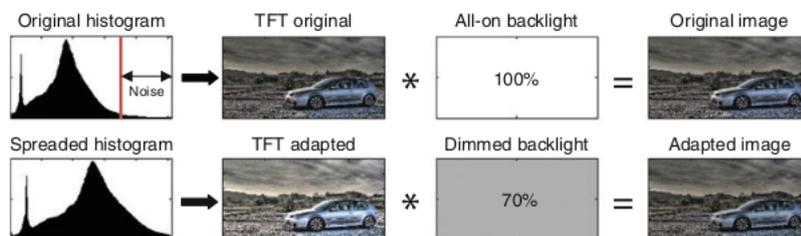


Abbildung 3.8.: Veranschaulichung einer histogrammbasierten Global-Dimming Methode. Entnommen aus [XAJ12].

Dies führt jedoch dazu, dass das ursprüngliche Bild stark geändert werden kann, zumal die Effekte des Dimmings bis hier noch nicht inbegriffen sind. Somit ist dieser Ansatz für aufgrund der hohen Requirements für den Automobileeinsatz weniger geeignet. Zudem wird ein Pixelwert festgelegt, ab dem die Pixel alle in Sättigung gehen (*Clipping*, siehe unten). Einen weiteren Ansatz liefert [Qib+15] durch eine Klassifizierung nach Kontrast und Helligkeit. In diesem Fall wird das Histogramm durch eine S-Kurve gespreizt und manipuliert dadurch ebenfalls die Bilddaten. Der entsprechende „dimming level“ welches dem Duty-Cycle der LED-Ansteuerung entspricht, wird in Abhängigkeit der Klassifizierung gesetzt und hängt nicht mehr direkt von den Pixelwerten ab. Dadurch kann jedoch eine falsche Klassifizierung zweier ähnlicher Bilder zu völlig unterschiedlichen Leuchtdichten führen.

Für die Umsetzung in Hardware wird der Algorithmus oft direkt in den TCON implementiert. In diesem Fall findet die Manipulation der Pixel durch eine adaptive Gammakurve statt, wie in Abbildung 3.9 gezeigt. Auf eine Pixelkompensation wird in diesem Fall oft verzichtet und das Bild wird dunkler. In der Abbildung, was dem SoA entspricht, wird das Gamma-Mapping im TCON abgeflacht. Diese Abflachung wird durch einen *Knickpunkt*  $Q$  bestimmt, welcher für die Bildqualität maßgeblich ist, da alle Grauwerte  $> Q$  manipuliert werden und wird in der Regel durch simple Logik bestimmt. Mit dieser Methode wird der ursprünglich maximale Grauwert nicht mehr erreicht, führt jedoch aber zur Reduktion des Strombedarfs (*Powersaving*). In diesem Beispiel betrüge das Powersaving 20 %, da der LED-Duty-Cycle auf 80 % eingestellt werden kann und berechnet sich wie in Formel (2.21) auf Seite 34 dargestellt.

Die Gammakurve aus Abbildung 3.9 kann mithilfe von  $Q$  angepasst werden, die Steilheit ( $WD$ ) der Abflachung der Gammakurve für Werte  $> Q$  ist in tatsächlichen Implementierungen konstant.

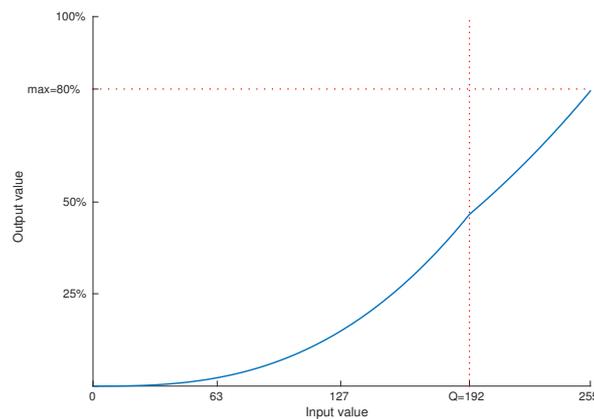


Abbildung 3.9.: Manipulation der Gammakurve zum Zweck des Global-Dimmings. Entnommen aus [SSX15].

Wegen der Einfachheit der SoA Algorithmen und deren günstigen Hardwareimplementierung kann das Bild jedoch schnell an Brillanz und Kontrast verlieren, da die Parametereinstellung wie beispielsweise der rein histogrammabhängige Knickpunkt  $Q$  ([SSM08]) immer einen Trade-Off darstellt und nicht für alle Inhalte gut geeignet sind, da sie verschiedene Charakteristik aufweisen können. Darunter kann auch die Ablesbarkeit bei Sonneneinstrahlung verschlechtert werden [Bau+10]. Eine Möglichkeit  $Q$  zu bestimmen, ist das Abzählen der e.g. 10 % der hellsten Pixel im Bild. Diese müssen dann rechts von  $Q$  liegen. Diese werden im Folgenden auch als  $PQ$  bezeichnet und entsprechen einer Anzahl der hellsten in Prozent. Anhand eines festen  $PQ$  ist jedoch eine ausreichend hohe Qualität bei gleichzeitigem hohen Powersaving nicht erreichbar.

Abbildung 3.10 zeigt ein typisches HMI Bild mit und ohne Global-Dimming. Es handelt sich um eine Fotoaufnahme auf dem in Abschnitt 3.4.3 vorgestellten Prototyp. Hier wurde nach SoA  $PQ = 10\%$

festgesetzt. Daraus wird dann der Knickpunkt  $Q$  berechnet. Das Powersaving beträgt hier 37 %, jedoch geht die Brillanz teilweise verloren und die Farbe wirkt weniger gesättigt, dadurch kann ein hochwertiges Displays als eins mit verminderter Qualität erscheinen und schlechter bei Sonneneinstrahlung ablesbar werden. Für ein natürliches Bild (vgl. Abbildung 3.1 auf Seite 42) kann die Qualität bei dieser Einstellung jedoch auch ausreichend sein [Sch+16b].

Ein neben dem Powersaving (siehe Formel (2.21)) wichtiges Kriterium zur Abschätzung eines Dimming Algorithmus ist das sogenannte *Clipping*. Die Ursache ist eine zu geringe Annahme der Leuchtdichte hinter den Pixeln sein, dadurch werden die bereits hohen Grauwerte in der Pixelkompensation an ihr Limit getrieben. Beispielsweise werden ab einem Kompensationsfaktor von  $\geq \frac{85}{81}$  alle 8 Bit Werte  $> 243$  auf den maximalmöglichen Wert 255 eingestellt (*geclippt*). Dadurch entsteht ein sichtbarer Verlust der Information.

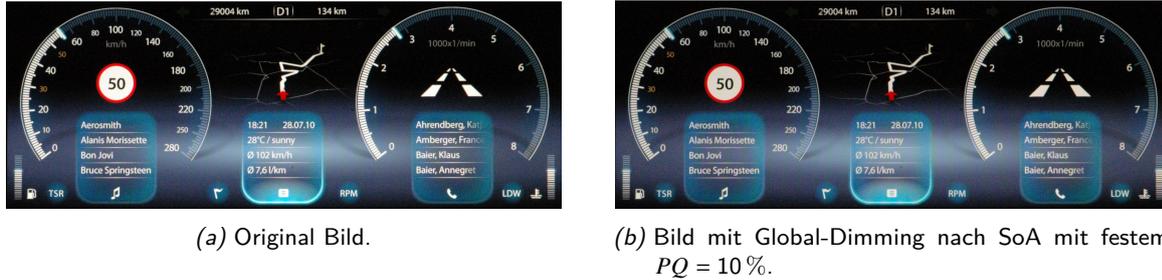


Abbildung 3.10.: Aufnahme (Fotokamera) eines HMI auf dem Prototyp. Entnommen aus [Sch+16b].

Insbesondere wenn einzelne Subpixel ins Clipping laufen, kommt es zu einer deutlich sichtbaren Farbverschiebung. Ein Beispiel dazu ist in Abbildung 3.11 abgebildet. Es wird ein Ausschnitt eines HMI Bildes gezeigt vor und nach der Pixelkompensation (siehe Abschnitt 2.5.4) durch einen Dimming Algorithmus. Neben dem Verlust von Information ist ein *Farbdrift* erkennbar. Die Histogramme mit 64 Bins repräsentieren die jeweils 256 Graustufen pro Subpixel. Nach der Kompensation ist die Verschiebung der Histogramme nach rechts, sowie das resultierende Mapping vormals verschiedener Grauwerte auf den maximal möglichen. Zum Beschreiben des Clippings eines Bildes wurden in [SJX12] und aufbauend in [Jun14] zwei Kennzahlen eingeführt. Dabei handelt es sich um die reine Anzahl der geclippten Pixel sowie einer mittleren Clipping-Magnitude, welche die Höhe des Informationsverlustes beschreibt. Sei nun  $p_{(i,j)} \in \mathcal{P}$  ein Pixel mit der Intensität  $P_{(i,j)}$  und die Backlightintensität  $\check{b}_{(i,j)}$  jeweils  $\in [0, \dots, 1]$ , dann sei die Menge der geclippten Pixel definiert als

$$\left\{ p_{i,j}, \frac{P_{(i,j)}}{\check{b}_{(i,j)}} > 1 \right\}$$

wobei die Indizes  $(i, j)$  die  $i$ -te Bildzeile und  $j$ -te Bildspalte beschreiben. Somit ergibt sich eine Clipping-Rate zu

$$C_N = \frac{\left| \left\{ p_{i,j}, \frac{P_{(i,j)}}{\check{b}_{(i,j)}} > 1 \right\} \right|}{|\mathcal{P}|} \quad (3.1)$$

Insgesamt wird zwar mit dem Global-Dimming Ansatz der Strombedarf etwas gesenkt, jedoch leidet die Bildqualität darunter, stellt jedoch einen guten Ansatz für eine kostenkünstige Methode für mobile Endgeräte im Consumerbereich dar. Nichtsdestotrotz ist diese Methode nicht automobiltauglich, da durch die starren Parametersätze das SoA Global-Dimming anfällig für Artefakte und zu aggressives Dimming ist.

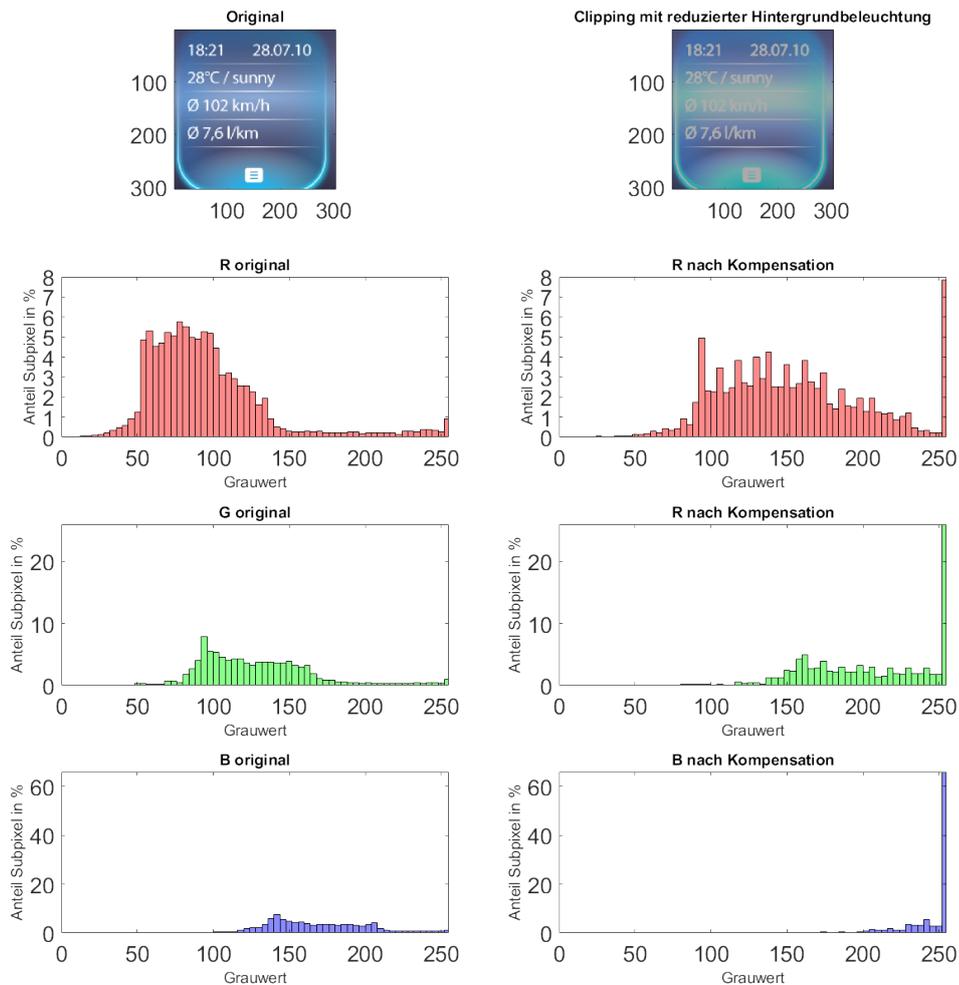


Abbildung 3.11.: Ausschnitt eines HMI Testbildes ohne (links) und mit Dimming (rechts). Das Clipping ist deutlich erkennbar. Darunter zeigen die Histogramme der einzelnen Subpixelfarben die Verteilung der 8 Bit Graustufen in einem Histogramm mit 64 Bins, sodass ein Bin eine Range von 4 Grauwerten zusammenfasst.

Die Bildqualität kann beträchtlich schwanken. Es kann Licht fehlen, sodass die Qualität degradiert wird und somit das Panel weniger hochwertig erscheint. Im Rahmen dieser Arbeit wird an einem speziell für automotive Displays geeigneten Global-Dimming Algorithmus gearbeitet. Weitere Bildcharakteristiken sollen den Knickpunkt  $Q$  berücksichtigen, um die Bildqualität zu erhöhen und eine gute Powersaving-Rate zu erhalten.

### 3.4. Verbesserung des Global-Dimmings zur Qualitätssteigerung

Im vorangegangenen Abschnitt wurde die Bildqualität als hauptsächliche Schwachstelle für die Anwendung eines Global-Dimming Algorithmus im Automobil identifiziert. Zudem darf die Powersaving Rate nicht zu gering sein um den Einsatz eines Global-Dimming Systems rechtfertigen zu können (hierbei sind die Hardwarekosten sehr relevant). In Abbildung 3.9 wurde der SoA dargestellt. Es existieren zum einen der Knickpunkt  $Q$ , ab dem die Kurve manipuliert wird sowie die Steigung nach dem Punkt  $Q$ , welches als  $WD$

bezeichnet wird. Im Rahmen dieser Arbeit wurde ein verbesserter Global-Dimming Algorithmus entwickelt, aus der eine gemeinsame Publikation mit *Continental* entstand [Sch+16b]. Zudem wurde ein Patent angemeldet, welcher diesen Global-Dimming Algorithmus schützt. [SSX16].

#### 3.4.1. Konzept

Der hier vorgestellte Ansatz soll das Bild hardwarechonend analysieren und abhängig davon die Parameter zur Steuerung einer indirekt einstellbaren Gammakurve bestimmen. Die Implementierung muss echtzeitfähig sein und auf der Pixelpipeline arbeiten. Das Ziel ist die adaptive Bestimmung der Anzahl der zu modifizierenden Pixel unter Berücksichtigung des Bildinhaltes. Daneben sollen Kunden- und Fahrerwünsche durch Einbeziehung von externen Parametern berücksichtigt werden können, um einen Trade-Off zwischen Powersaving und Erhöhung der visuellen Qualität zu erzeugen. Somit wird ein guter Kompromiss zwischen Komplexität und Nutzen erwartet.

Eine Möglichkeit die Gammakurve zu manipulieren, ist die Manipulation der Eingangsdaten in der wahrnehmungslinaren Domain wie in Abbildung 3.12 visualisiert. Diese ergeben dann durch Anwendung einer Gamma-Tabelle eine geänderte Gammakurve wie sie in Abbildung 3.9 zu sehen ist.

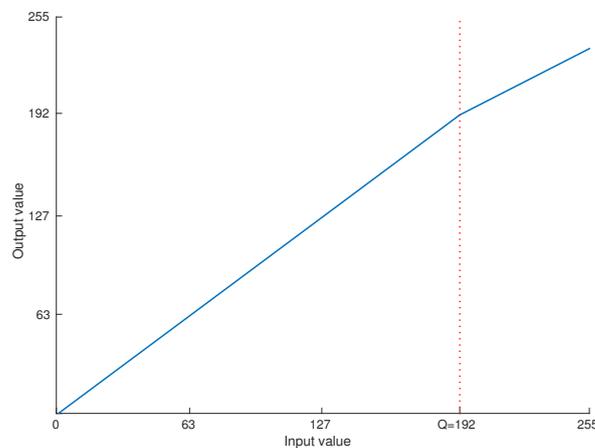


Abbildung 3.12.: Manipulation Eingangswerte basierend auf wahrnehmungslinaren Daten. Entnommen aus [SSX15].

Die Echtzeitanalyse auf der Pixelpipeline soll also die charakteristischen *Features* eines Bildes extrahieren und mit diesen Informationen eine adaptive Anzahl an Pixel festlegen, welche *ohne* negative Auswirkung auf die visuelle Qualität des Bildes zugunsten des resultierenden Strombedarfs der Hintergrundbeleuchtung modifiziert werden dürfen. Dies wird unter anderem dadurch ermöglicht, dass eine geringfügige Änderung in hellen Bereichen nicht immer als deutlicher Unterschied wahrnehmbar sind. Somit besteht hier noch Potenzial, um den Strombedarf (insbesondere im Tagbetrieb) ohne sichtbare Qualitätsverluste reduzieren zu können. Der Schwarzwert wird durch das Global-Dimming ebenfalls verbessert, da das Backlight gedimmt wird. (siehe Abschnitt 2.4). Dies ist insbesondere für die wahrgenommene Qualität im Nachtbetrieb wichtig. Das Blockdiagramm des vorgestellten Algorithmus ist in Abbildung 3.13 abgebildet und wird direkt in den Bilddatenstrom eingefügt. Er besteht aus drei Phasen, der *Preprocessing*, der *Dimming Algorithmus* sowie der *Postprocessing* Phase. Die Pixelpipeline ist als roter Datenpfad markiert. Als Ergebnis werden neben dem prozessierten Bilddatenstrom auch der berechnete PWM-Wert bzw. Duty-Cycle zur Ansteuerung der Hintergrundbeleuchtung ausgegeben.

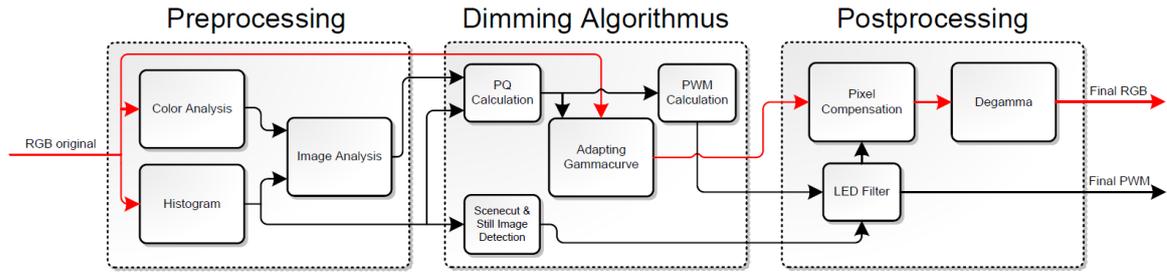


Abbildung 3.13.: Blockdiagramm des vorgestellten Global-Dimming Algorithmus. Entnommen aus [Sch+16b].

Die *Preprocessing* Phase hat das Ziel das Bild global zu analysieren und die Features abzuleiten. Dazu wird ein Histogramm des Bildes generiert und für die Auswertung gespeichert (siehe Abschnitt 3.4.2). Diese Auswertung trägt in der darauf folgenden Phase zur Entscheidungsfindung bei wie sehr (*WD*) und ab welchem Knickpunkt (*Q*) die Gammakurve abgeflacht werden kann, um die Qualität noch zu erhalten, aber den Strombedarf reduzieren zu können. Dabei wird der Wert *PQ* in der Phase *Dimming Algorithmus* adaptiv bestimmt, von dem aus sich der eigentliche Knickpunkt *Q* berechnen lässt (wurde weiter oben vorgestellt). *PQ* ist also eine Funktion von den extrahierten *Features* wie weiter unten gezeigt wird. Basierend darauf werden wie in Abbildung 3.12 gezeigt, zunächst die Bilddaten im wahrnehmungslinearen Bereich anhand des berechneten Knickpunktes *Q* und mit der Steigung *WD* angepasst, bevor die Gamma-LUT durchlaufen wird. Somit berechnen sich der modifizierte Grauwert  $P'_i$  an linearisierter Pixelposition *i* hardwarechonend wie folgt

$$P'_i = \begin{cases} P_i, & \text{wenn } P_i < Q \\ \frac{WD \cdot P_i + (2^N - WD) \cdot Q}{2^N}, & \text{sonst} \end{cases} \quad (3.2)$$

Dabei entspricht  $P_i \in \{R_i, G_i, B_i\}$  den 8 Bit Grauwerten der aktuellen Subpixel im Bilddatenpfad. Die Quantisierung der Stufen beträgt hier  $2^N$  und  $WD \in \{0, \dots, 2^N\}$ . In der Implementierung wird  $N = 4$  verwendet und liefert folglich 16 Stufen als Gewichtung zwischen *P* und *Q* und kann durch einen Parameter gesetzt werden. Für  $WD = 2^N$  gilt  $P'_i = P_i$  und bei  $WD = \frac{2^N}{2}$  gilt  $P'_i = \frac{P_i + Q}{2}$ . Der LED-Wert berechnet sich dann mit

$$\mathbf{v} = (\max P'_i)^y, \quad \text{mit } i \in \{0, \dots, N_P - 1\} \quad (3.3)$$

Beim *Postprocessing* werden die modifizierten Bilddaten kompensiert. Die Kompensation kann dabei *dynamisch* auf dem temporal gefilterten LED-Wert oder *statisch* auf dem berechneten LED-Wert basieren. Die Kompensation selbst wird wie in Abschnitt 2.5.4 eingeführt (siehe auch [SJX12]), implementiert und ein inverses Gamma Mapping angewandt. Ohne Kompensation würde das *gedimmte* Bild dunkler. Um eine Farbverschiebung zu vermeiden, wird während der Kompensation sichergestellt, dass das Verhältnis der einzelnen Primärfarben erhalten bleibt [Sch+16b; Sch17].

Im folgenden Abschnitt wird nun auf die Methoden zur Extraktion der Features eingegangen, auf denen die Bildanalyse basiert.

### 3.4.2. Globale Bildanalyse

Um das Ziel der kostengünstigen Realisierung erreichen zu können, werden Features genutzt, welche teilweise aus globalen Histogrammen extrahiert werden können. Eine lokale Betrachtung der Features ist für Global-Dimming nicht zielführend und hat ein schlechtestes Kosten/Nutzen-Verhältnis, da hier *global* gedimmt wird. Für die Bestimmung von  $Q$  oder  $WD$  werden die folgenden Merkmale extrahiert und zur Analyse wie in [SSX15; Sch+16b; SSX16] vorgestellt, verwendet. Eine sehr wichtige Bedeutung im Automobil hat die wahrgenommene, subjektive Brillanz aber auch die Farbtreue.

In Abbildung 3.14 ist ein beispielhaftes Histogramm  $H$  mit Markierungen dargestellt. Diese dienen der Veranschaulichung bei der Einführung der Features und basiert auf den maximalen Subpixeln.

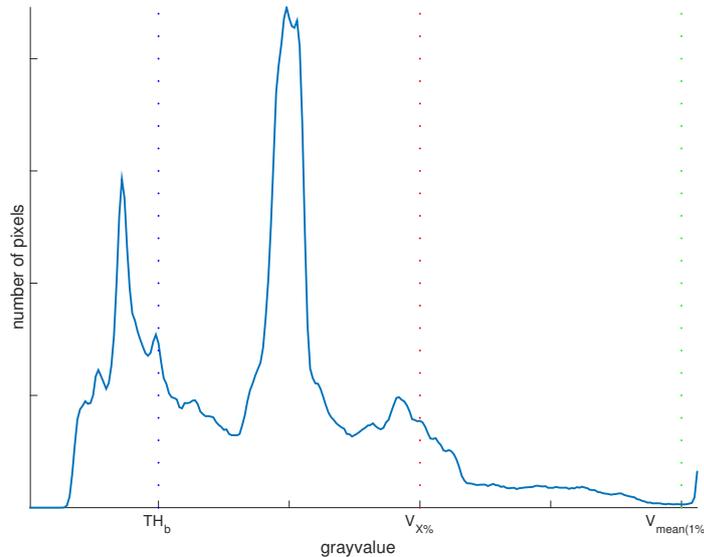


Abbildung 3.14.: Ein Histogramm eines Testbildes zur Veranschaulichung der relevanten Größen für das histogrammbasierte Global-Dimming. Entnommen aus [SSX15].

#### Darkness

Aufgrund der hohen Empfindlichkeit bei dunklen Bildinhalten [Bau+10] wird das sogenannte *Darkness* Feature extrahiert aus dem Histogramm extrahiert und kann beschrieben werden als  $e_1 = F_1(TH_b, H)$  und stellt eine Funktion von einem Threshold  $TH_b$  sowie des Histogramms  $H$  des Bildes dar (vgl. obige Abbildung).

#### Brillanz

Da im Allgemeinen ein Bild *brillant* erscheint, wenn nur wenige helle Objekte von größeren schwarzen Flächen umgeben sind, wird eine Metrik zur Bestimmung eines globalen Brillanz eingeführt. Diese ist eine Funktion  $e_2 = F_2(e_1, H(255))$  welche von dem Darkness Feature, sowie dem Histogramm an der Stelle 255 abhängt. Es wird ein Histogramm mit 256 Bins angenommen.

### Verhältnis der hellsten Pixel

Ein weiteres Feature betrachtet das Verhältnis zwischen dem Durchschnittswert der hellsten 1% der Grauwerte ( $V_{\text{mean}(1\%)}$ ) und dem Wert ab dem X% der Werte ( $V_{X\%}$ ) im Histogramm liegen. Beim ersteren wird ein Median genommen, um den Einfluss von Pixelrauschen in Bildern zu reduzieren. Dieses Feature ist ebenfalls ein Indikator für die Brillanz und globalen Kontrast des Bildes. Es wird mit  $e_3 = F_3(V_{\text{mean}(1\%)}, V_{X\%})$  beschrieben. Außerdem ist aus der Wahrnehmung bekannt, dass eine bestimmte Anzahl der hellsten Pixel ohne wahrnehmbare Verschlechterung der Bildqualität *geclippt* werden kann.

### Farbigkeit

Als weiteres Feature wird ein Maß für die Farbsättigung genutzt. Denn gesättigte und helle Farben benötigen Hintergrundlicht, um nicht blass zu wirken und um einen Farbdrift zu vermeiden. Dieses Feature benötigt anders als die vorher genannten keine Werte aus dem Histogramm, sondern extrahiert die nötigen Daten direkt aus der Pixelpipeline. Das Feature wird als  $e_4 = F_4(R_i, G_i, B_i)$ , mit  $i \in \{0, \dots, N_P - 1\}$  beschrieben.

Die vorgestellten Features werden nach der Berechnung in einen repräsentativen und charakteristischen Wert bzw. Maß zusammengefasst. Er wird hier auch als *Criticality Degree*  $BD$  bezeichnet und berechnet sich als gewichtete Superposition der extrahierten Features wie folgt (nach [SSX15; Sch+16b])

$$BD = \sum_{i=1}^4 w_i \cdot e_i \text{ mit } w_i \in \mathbb{R}$$

Dabei entsprechen die  $w_i$  den Gewichten für die einzelnen Features. Dieser repräsentative Wert  $BD$  kann dann dem prozentualen Anteil der *modifizierbaren* Pixel  $PQ$  sowie dem Wert  $WD$  (siehe Formel (3.2)) zugeordnet werden. Diese entsprechen dann  $PQ = L_1(BD)$  sowie  $WD = L_2(BD)$ . Die *Mappings*  $L_1$  und  $L_2$  können durch LUTs realisiert werden.

Ein Beispiel für das Mapping auf  $PQ$  ist in Abbildung 3.15 dargestellt. Es sind jeweils die Thresholds  $TH_{nc}$  (unkritisch) und  $TH_{vc}$  (sehr kritisch) markiert. Folglich ist für ein  $BD \geq TH_{vc}$  das  $PQ = 0\%$  und somit eine Manipulation der Pixel zwecks Reduktion des Strombedarfs nicht möglich, da die Bildqualität ansonsten degradiert werden würde. Für den Fall  $BD < TH_{nc}$  wird das Bild als unkritisch erachtet und der Prozentsatz kann der maximal zulässige sein ( $PQ_{\text{max}}$ ). In der Implementierung ist er zu  $PQ_{\text{max}} = 13\%$  gewählt. Zwischen  $TH_{nc}$  und  $TH_{vc}$  kann der prozentuale Anteil der manipulierbaren Pixel  $PQ$  linear abfallen, wie in der Abbildung 3.15 dargestellt ist. Alternativ kann diese Funktion auch detaillierter angepasst werden. So wird in der realen Implementierung die Steigung für  $BD \rightarrow TH_{vc}$  betragsmäßig geringer.

Mit dieser Methode wird für ein kritisches Bild die Bildqualität (farbenfrohe und als brillant wahrgenommene Bilder) erhalten und erlaubt andererseits den Strombedarf bei unkritischen Bildern stärker zu senken. Im Folgenden wird näher auf die Realisierung eingegangen.

### 3.4.3. Hardwareimplementierung

Die Anforderungen an Displays für automobiler Anwendungen sind wie in Abschnitt 3.1 beschrieben sehr hoch. Ein Dimming-System darf keine hohen Latenzen der Bilddaten verursachen und gleichzeitig kostengünstig umgesetzt werden können. Zur Implementierung der Global-Dimming IP wurde ein FPGA der Firma *Xilinx* verwendet. Es handelt sich hier bei um das Spartan 6 FPGA, welches mit 45 nm Strukturweite gefertigt wurde. Genauer ist die Ausführung XC6SLX150T-3FG900, welches über 540 nutzbare I/Os zur Verfügung

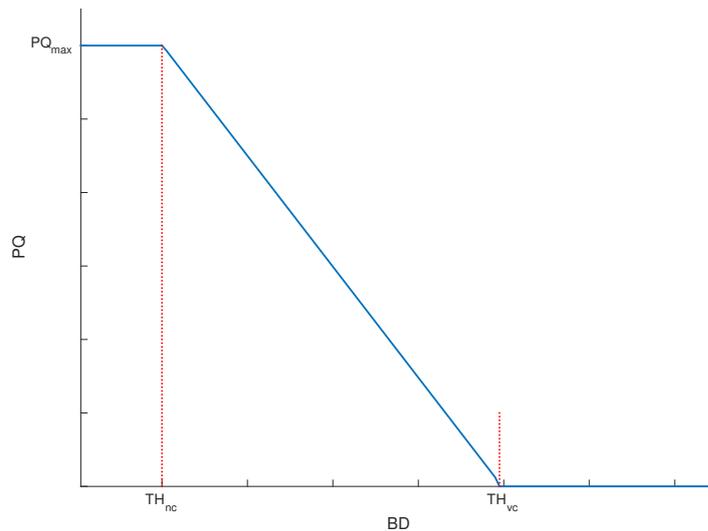


Abbildung 3.15.: LUT für eine bildabhängige Zuordnung von dem *criticality degree*  $BD$  auf  $PQ$  (Anzahl der hellsten Pixel in Prozent). Entnommen aus [SSX15].

stellen kann und umgerechnet 147443 *Logic-Cells* (basiert auf 6-Eingang LUTs) beinhaltet [Xil11]. Es finden auf dem FPGA 23038 Slices mit jeweils 4 LUTs und 8 FFs Platz. Insgesamt verfügt dieser Spartan 6 über 184304 FFs sowie 4,824 MBit Block-RAM.

Zur Konnektierung der nutzbaren I/Os des FPGAs verfügt das FPGA-Board über mehrere FPGA Mezzanine Cards (FMCs) und zwar 1 HPC mit 400 Pins und 2 LPC mit 160 Pins (darunter auch GND und VCC Pins). In der weiter unten folgenden Abbildung 3.17 wird das FPGA-Board mit weiteren Printed Circuit Boards (PCBs, Platine, gedruckte Schaltung) gezeigt.

In Abbildung 3.13 wurde der Algorithmus in einem Blockdiagramm dargestellt. In diesem Abschnitt wird auf die tatsächliche Hardwarerealisierung eingegangen. Die Architektur basiert auf der Pixelpipeline und ist echtzeitfähig. Das *Timing* innerhalb eines Frames spielt eine besonders wichtige Rolle in einer Hardwareumsetzung, sodass die Gegebenheiten wie Displayauflösung und Bildwiederholraten mit berücksichtigt werden müssen. In Abschnitt 2.6 wurde unter anderem die Pixelpipeline und der zeitliche Ablauf (Display-Timing) vorgestellt.

#### Frame-Timing

Die Bilddaten eines Bildes (*frame i*) werden also erst alle gelesen worden sein, nachdem ein Bild durch die Pixelpipeline übertragen wurde. Somit muss sich der daraus berechnete PWM-Wert für die Hintergrundbeleuchtung immer auf das vorangegangene Bild *frame i-1* beziehen. In Abbildung 3.16 wird dies dargestellt. Das aktuelle, aktive Bild *frame i* befindet sich in der Mitte der Abbildung. Die roten Pfade markieren den Weg der Pixelpipeline durch die einzelnen Module. Es wird also während Bild (Bilddaten) *i* aktiv über die Pixelpipeline läuft, das Histogramm und das Farb-Feature berechnet. Darauf aufbauend, wird in der vertikalen *Blanking*-Zeit der Knickpunkt  $Q$  bestimmt (basieren auf  $PQ$ ). Dieser wird dann wiederum im nächsten Bild *i+1* zur Anpassung der Gammakurve genutzt und der zum vorherigen Bild berechnete PWM-Wert wird für die Pixelkompensation genutzt.

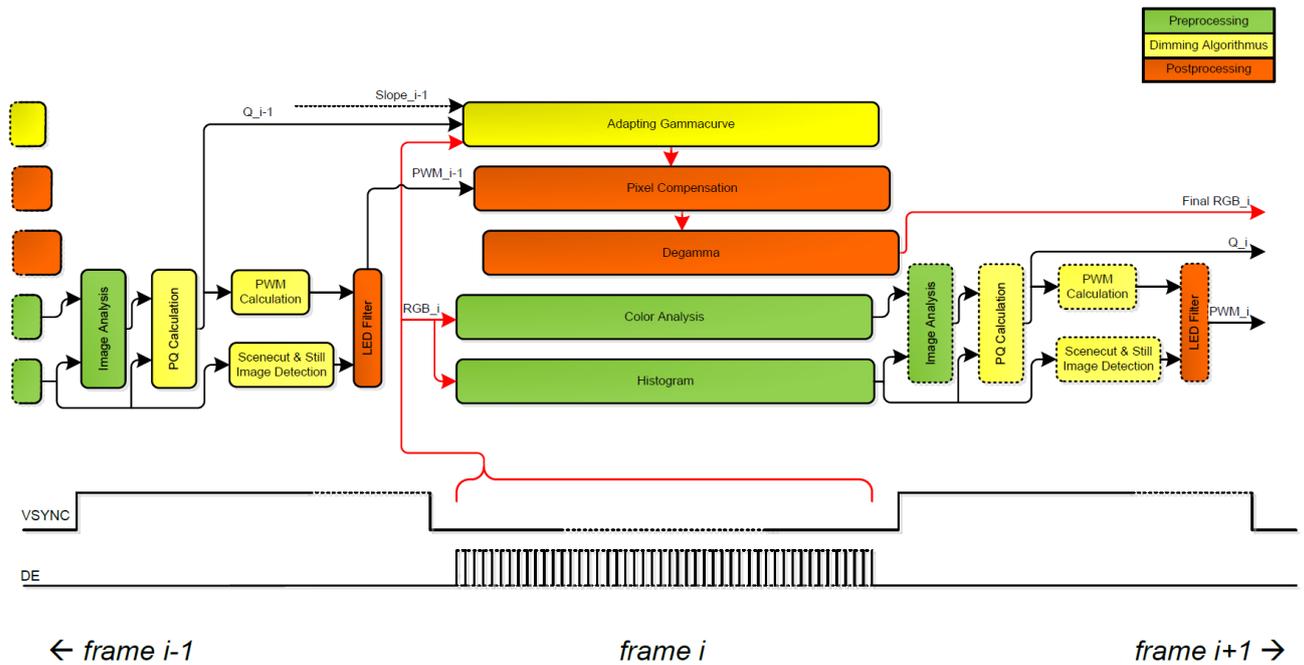


Abbildung 3.16.: Zeitlicher Ablauf des Global-Dimming Algorithmus in Hardware. Die markierten Blöcke entsprechen den aus Abbildung 3.13.

### Prototyp

Diese Implementierung basiert auf einem typischen automotive 12,3-Zoll ICDs mit einer Auflösung von  $540 \times 1440$  Pixeln. Bei einer Framerate von 60 Hz, beträgt die Pixelclock hier 62,33 MHz.

Neben des eigentlichen FPGA Boards, wurde noch eine PCB für den LVDS-Ausgang genutzt, sowie eine eigene PCB für den HDMI Eingang (TMDS wird ins FPGA weitergeleitet). Zudem wurden in einem einzigen FPGA zwei autarke Global-Dimming Prozessoren in Verilog implementiert. Sie hatten einen gemeinsamen Bildeingang (HDMI), jedoch separate LVDS Ausgänge. Somit ist es möglich direkt zwei Display exakt synchron ansteuern und zwei verschiedene Parametersätze des Global-Dimming Algorithmus im Direktvergleich sehen zu können. Der FPGA-Aufbau ist in Abbildung 3.17 dargestellt, während in Abbildung 3.18a zwei synchron angesteuerte LCDs gezeigt werden, welche sich direkt übereinander befinden. Dies ermöglicht einen guten Direktvergleich zweier verschiedene Paramatersätze des Global-Dimming Algorithmus. In Abbildung 3.18b wird die Auswahl für die Anzeige verdeutlicht. Die Parameter lassen sich über Universal Asynchronous Receiver Transmitter (UART) einstellen, während einige Einstellung auch direkt über Schalter am FPGA Board realisiert wurden. Das Panel 2 kann hier zwischen einer gedimzten und einer ungedimzten Ausgaben umgeschaltet werden und kann hier als Referenz dienen.

### Hardwarekosten

Aufgrund der effizienten Implementierung, konnten zwei autarke Global-Dimming Prozessoren in einem einzigen FPGA realisiert werden. Ein Prozessor benötigt hier lediglich 207 kBit des Block-RAMs und umgerechnet  $\approx 17000$  Logic Cells [Sch+16b]. Die Pixel werden in der Pixelpipeline prozessiert und somit wird auf einen Frame-Buffer verzichtet. Mit der oben angegebenen Pixelclock entsteht ein Pixeldelay

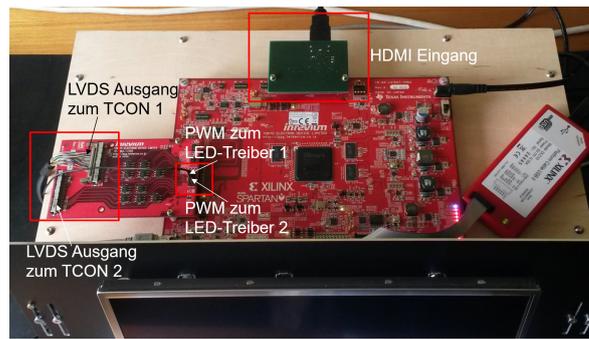
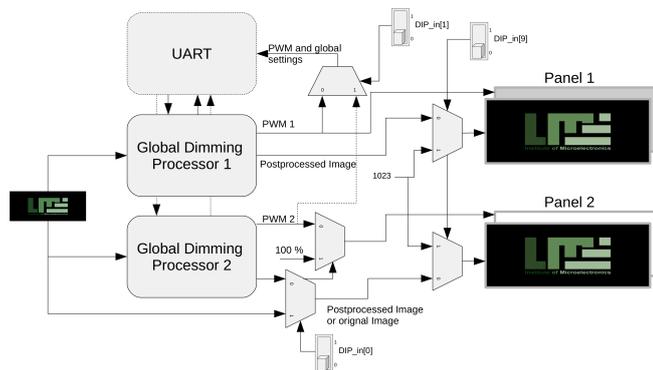


Abbildung 3.17.: Anschluss der Prototypen an das FPGA Board. Das FPGA selbst ist der große IC in der Mitte des PCBs. In diesem Bild ist das zweite Display nicht angeschlossen.



(a) Zwei Prototypendisplay, angesteuert durch ein einziges FPGA, jedoch mit autarken Global-Dimming Prozessoren.



(b) Einstellungs- und MUX-Möglichkeiten der beiden implementierten Global-Dimming Instanzen im FPGA.

Abbildung 3.18.: Prototypenaufbau aus zwei Displays zum direkten Vergleich. Zwei unabhängige Global-Dimming Prozessoren sind in einem einzigen FPGA implementiert. Das untere Display lässt sich gedimmt als auch ungedimmt für einen direkten Vergleich betreiben.

vom Eingang zum Ausgang des FPGAs von nur 641,74 ns, welches einem Delay von 40 Pixelclock-Cyclen entspricht. Die Kosten in einem TCON Chip liegen im Cent-Bereich.

#### 3.4.4. Ergebnisse

Der hier vorgestellte Global-Dimming Algorithmus, welcher im Rahmen dieser Arbeit entwickelt wurde, lässt sich wie gezeigt wurde sehr hardwarechonend realisieren. Von dem genutzten Spartan 6 FPGA wurden lediglich 11,5% der *Logic Cells* genutzt. Die wahrgenommene Bildqualität ist gegenüber dem ungedimten Modus weitgehend erhalten, während das Powersaving aufgrund der fehlenden Lokalität moderat ausfällt. Das Resultat der automatischen *PQ* Auswahl wird in Abbildung 3.19 dargestellt. Dazu wurde das Powersaving für das International Electrotechnical Commission (IEC, Internationale Elektrotechnische Kommission)-Testvideo [DIN] gemessen. Es wurde der Parameter *WD* auf der Abszisse variiert. Die Ordinate zeigt das Powersaving in Prozent an. In Rot und Gelb sind jeweils zwei konstante *PQ* aufgetragen, während dem blauen Verlauf eine automatische *PQ* Berechnung zugrunde liegt. Es ist zu erkennen, dass sie sich zwischen den sehr konservativen  $PQ = 2\%$  und dem im Durchschnitt sehr qualitätsdegradierenden  $PQ = 8\%$  befindet und

somit hier einen Trade-Off darstellt. Das Powersaving wird im Allgemeinen für  $WD \rightarrow 1$  immer geringer, da sich die resultierende Gammakurve sich immer weiter der Ursprungskurve nähert. In Abbildung 3.20 werden

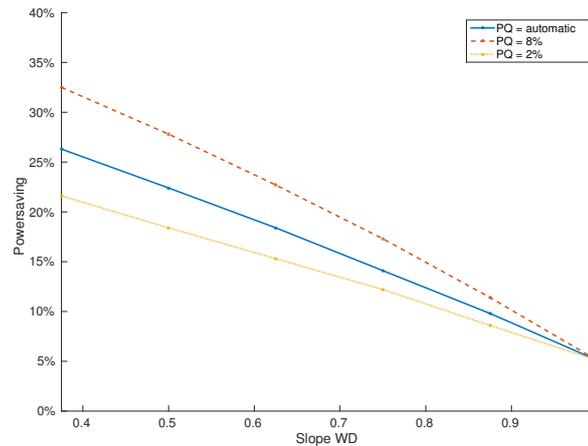


Abbildung 3.19.: Powersaving mit variiertem  $WD$  bei festen und automatischem  $PQ$  basierend auf dem IEC-Testvideo.

zwei Aufnahmen gezeigt welche die Qualitätserhaltung durch die Bildanalyse beispielhaft demonstriert. Links handelt es sich um SoA Global-Dimming, während rechts der hier vorgestellte Algorithmus zum Einsatz kam. In dem markierten Bereich ist beim SoA Ansatz ein deutlich sichtbares Clipping zu sehen, während rechts die Qualität erhalten bleibt. Die Performance des Powersavings wird anhand eines zum Zeitpunkt der Arbeit



Abbildung 3.20.: SoA im Vergleich zum hier vorgestellten Global-Dimming Algorithmus anhand eines Bildausschnitts. Entnommen aus [SSX15].

an dem Global-Dimming System vorhandenen Testsets in Abbildung 3.21 dargestellt. Die Bilder wurden hier nach mittlerer Leuchtdichte vor dem Dimmen auf der Abszisse sortiert (rot). Die mittlere Leuchtdichte nach dem Dimmen ist in Blau aufgetragen sowie der korrespondierende Leistungsumsatz in Schwarz. Es ist sehr gut erkennbar, dass die mittlere Leuchtdichte nur minimal von der ungedimmten Variante abweicht. Dabei wird für dieses Testset ein Powersaving von 12% erreicht. Es wird ersichtlich, dass die mittlere Leuchtdichte nicht mit dem Powersaving korreliert, da hier das Bild anhand der Features bewertet wird und das Resultat tatsächlich inhaltsabhängig wird. Somit wird bestätigt, dass das Dimming in Abhängigkeit von Bildmerkmalen wie Brillanz und Farbigkeit, etc. erfolgt und unabhängig der mittleren Leuchtdichte. Die visuelle Qualität für die kritischen Bilder wird somit erhalten. Es existieren jedoch auch Bilder, welche von dem Algorithmus nicht gedimmt werden können, um eine Qualitätsdegradation zu vermeiden. Abbildung 3.22 zeigt einen Vergleich der Leuchtdichtemessungen (durchgeführt mit einer Eldim UMaster) des ungedimmten und des gedimmten Zustandes. Ein Powersaving beträgt 9,4% während die Leuchtdichte sehr nahe am ungedimmten Stand heranreicht.

### 3.4. Verbesserung des Global-Dimmings zur Qualitätssteigerung

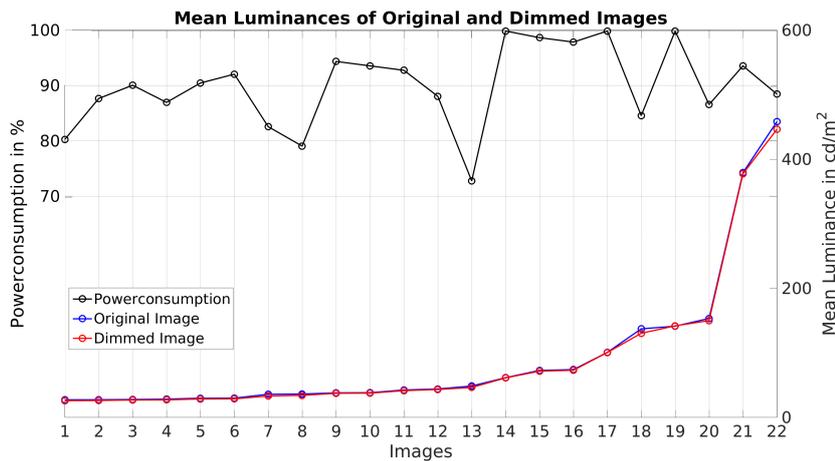


Abbildung 3.21.: Vergleich der mittleren Leuchtdichten (Testbilder aufsteigend sortiert) für gedimmte und ungedimmte Testbilder, sowie des Leistungsumsatzes. Aufnahmen durchgeführt mit der Eldim UMaster. Entnommen aus [Sch+16b].

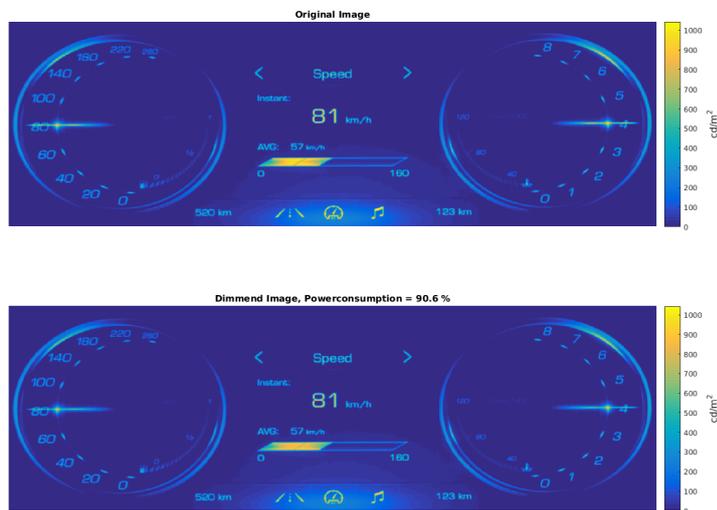


Abbildung 3.22.: ELDIM Leuchtdichte-Messung. Oben ungedimmt, unten gedimmt mit einem Powersaving von 9,4 %.

### Zusammenfassung

Es wurde mit der hier vorgestellten Global-Dimming Methode gezeigt, dass eine in Hardware kostengünstige Lösung basierend auf der Pixelpipeline möglich ist. Die Echtzeitanforderungen werden erfüllt und eine Beeinträchtigung bei der Darstellung von sicherheitsrelevanten Inhalten wird durch die Bildanalyse vermieden. Diese Methode eignet sich durch ihre Robustheit für den Automobilbereich und ist universell und unabhängig vom Displaymodell einsetzbar.

Zudem wurde in einer am LME angefertigten Masterarbeit von Werner der hier vorgestellte Algorithmus auf einer GPU implementiert [Wer16] und somit die Möglichkeit der Portierung aufgezeigt. Dabei wurde ein NVIDIA Tegra K1 System-on-Chip genutzt. Die Implementierung wurde in der OpenGL Shading Language (GLSL) auf den Shadern durchgeführt [SA14]. Diese sind in Shader-Stufen organisiert und stellen eine GPU

Pixelpipeline dar.

Die bisherigen Erkenntnisse der Global-Dimming Methode mündeten zudem auch in didaktische Resultate. Die Entwicklung eines Global-Dimming Algorithmus wurde als Praktikumsveranstaltung für Masterstudenten der Universität des Saarlandes umgesetzt, voranging für die Studenten der Naturwissenschaftlich-Technischen Fakultät sowie der Fakultät für Mathematik und Informatik. Ein Script wurde dafür ebenfalls neu aufgesetzt [Sch+19a]. Die Global-Dimming Methode ist dabei stark abstrahiert um die Komplexität zu reduzieren und die Studenten auf die Problemstellung zu sensibilisieren. Dabei wird mithilfe der Hardwarebeschreibungssprache Verilog das Konzept der Pixelpipeline erarbeitet und schließlich im FPGA realisiert. Der Erfolg ist direkt auf einem LCD sichtbar. Somit bekommen Studenten die Gelegenheit sich mit LCDs zu beschäftigen und die Probleme, wie in diesem Fall der hohe Strombedarf zu verstehen und Lösungen zu entwickeln.

Neben den Vorteilen wie günstige Hardwarerealisierung existieren jedoch auch Schwächen. Wie bereits dargestellt wurde, gibt es kritische Bildinhalte, für die der Strombedarf nicht reduziert werden kann. Dies trifft meistens kontrastreiche und brillante Bilder. Werden die Parameter jedoch so eingestellt, dass trotzdem der Strombedarf reduziert werden soll, so kann es oft zu Artefakten wie *Clipping* kommen. Der Grund liegt an sehr hohen Grauwerten, die in einem künstlichen HMI fast immer vorkommen. Eine Verbesserung dieser Situation verspricht der Einsatz eines Local-Dimming Algorithmus, da dann mehrere unabhängige LEDs für die Ausleuchtung verfügbar sind und somit auch für solche kritische Bilder der Strombedarf reduziert werden kann.

### 3.5. Local-Dimming

Im vorangegangenen Abschnitt wurde eine kostengünstige Global-Dimming Lösung vorgestellt, welche jedoch nicht immer ein hohes Powersaving liefern kann, selbst wenn die mittlere Leuchtdichte eines Bildes gering ist, wie Abbildung 3.21 zeigt. Eine bessere Performance bzgl. des Powersavings wird von einem Local-Dimming System erwartet, da durch das Dimmen einzelner LED-Stränge eine von der BLU abhängige *Lokalität* erreicht werden kann. Folglich gab es erste Prototypen eines High Dynamic Range (HDR, hoher Dynamikumfang) Systems von Seetzen u. a. bereits im Jahre 2003 [SWW03; See+04]. Das Interesse von Displayherstellern wurde durch die Vorstellung eigener Local-Dimming Algorithmen deutlich [Yeo+08; Che+06] (*Samsung*). Zudem wurden weitere Veröffentlichungen von Local-Dimming Systeme publiziert wie [Shi+06; Hd09]. Der Nachteil dieser Systeme war der hohe Berechnungsaufwand, welcher in Hardware in diesen Umfängen nicht kostengünstig umsetzbar war. In einigen Publikationen wurde eine Entfaltung vom Bild auf die einzelnen LED-Werte suggeriert [Li+07]. Zudem setzen diese Algorithmen auf Methoden zur Reduktion der Bildauflösung und haben keinen physikalischen Bezug. Folglich sind sie nicht frei von Artefakten wie Clipping. Beispielsweise wird in [GH07; Zha+19] eine Histogrammanalyse genutzt und kann folglich artefaktbehaftet sein.

Zudem wurde der Crosstalk zwischen den LEDs nicht oder nicht physikalisch berücksichtigt, wie Abbildung 3.7 auf Seite 46 aus [Wan+17] zeigt. Es wurde ein Benchmark vorgestellt, welcher die maximale Leuchtdichte bei einem OLED sowie bei einem LC-Display mit vielen LEDs mit Local-Dimming bei immer größer werdender weißer Fläche abbildet. Es ist zu erkennen, dass nur ein Teil der maximalen Leuchtdichte durch dieses Local-Dimming System erreicht wurde. Insbesondere für automobiler Anwendungen ist dieses Verhalten des Systems nicht hinnehmbar. Die Ursache ist dem nicht berücksichtigten Crosstalk geschuldet.

Diese Missstände wurden durch den SSC-Algorithmus gelöst und erfolgreich für Edge-Lit BLUs für maximal

24 einzeln ansteuerbaren LED-Stränge in Hardware basierend auf der Pixelpipeline validiert. Die Grundlagen des SSC Verfahrens wird in den nächsten Unterabschnitten vorgestellt.

#### 3.5.1. Edge-Lit Dimming (SSC) und Stand der Technik

Da diese Arbeit auf einigen Methoden des SSC Verfahrens aufbaut, werden hier dessen Grundlagen als Stand der Technik eingeführt. Das Verfahren wurde für Edge-Lit BLUs erfolgreich für Consumer-Geräte validiert und ist in der Fachwelt bekannt.

Der Algorithmus und seine Erweiterungen wurden auf vielen internationalen Konferenzen und Journalen publiziert [Jun+11; Grü+16; Grü+19; Alb+10; AKX08; Jun+13; SJX12; Sch+16a; SSX15; Jun+13; AKX09] und Abschlussarbeiten am LME angefertigt wie beispielsweise [Sch13; Jos20]. Zudem wurde in mehreren Arbeiten an dem SSC und aufbauende Problemstellungen geforscht. Albrecht entwickelte und erforschte die Grundlagen des SSCs Algorithmus [Alb10]. Jung behandelte in [Jun14] die visuelle Qualität des Local-Dimmings bei Edge-Lit LCDs und Schäfer entwickelte neue Methoden zur Qualitätssicherung von Local-Dimming [Sch17]. In den bisherigen Arbeiten stand die Darstellung von natürlichen Bildern auf Consumer Geräten wie TVs im Mittelpunkt. Die grundlegenden Konzepte des SoA werden auch mit den Patenten [XKA13; Xu+19] beansprucht.

Die Abkürzung SSC steht wie in der Einleitung erwähnt für Sorted Sector Covering und hat das Ziel durch eine iterative Methode sich einer reproduzierbaren Lösung anzunähern. Es soll also das in Formel (2.22) auf Seite 34 beschriebene Optimierungsproblem gelöst werden. Dabei wird die in Abschnitt 2.5.2 eingeführte Einflussmatrix verwendet. Der Abkürzung leitet sich aus den einzelnen Methoden des Algorithmus ab.

*Sorted* steht dabei für eine vorgegebene Abarbeitungsreihenfolge der Pixel. Diese sind anhand der Einflussmatrix so sortiert, dass diejenigen Pixel priorisiert werden, welche die höchsten Lichtbeiträge einer LED erfahren. In der Sortierung wird ebenfalls berücksichtigt, dass zwei Pixel welche nacheinander prozessiert werden *nicht* dieselbe dominante LED aufweisen dürfen, also in verschiedenen *Sektoren* liegen. Mit der dominanten LED ist diejenige LED gemeint, welche den höchsten Einfluss auf ein bestimmtes Pixel hat. Sie wird im später auch als *Main-LED* oder *Haupt-LED* bezeichnet. Der Begriff *Covering* beschreibt hier das Constraint aus Formel (2.22) (Seite 34), welches die erforderliche Mindesthelligkeit hinter einem bestimmten Pixel fordert. Dabei wird iterativ vorgegangen, sodass der Crosstalk der LEDs auf in einem Pixel sukzessive berücksichtigt werden kann.

Durch eine Sortierung nach Einflüssen, also der Lichtbeiträge wird ein Folgezustand in der Prozessierung vorgegeben, welcher sich der optimalen Lösung am besten nähert. Dabei ist es sinnvoll zunächst die Pixel zu prozessieren, welche einen hohen Einfluss von einer LED erfahren und zum Schluss diejenigen mit einem geringeren LED-Einfluss. Durch Sortierung wird sichergestellt, dass alle Pixel das Constraint der geforderten Helligkeit erfüllen können. Zudem wird das Powersaving maximiert, da die LEDs durch die Crosstalk-Berücksichtigung bereits genug Licht hinter einem Pixel erzeugen können. Denn eine LED mit wenig Einfluss müsste einen höheren Duty-Cycle liefern, um einen Pixel abzudecken, als eine mit einem höheren Einfluss auf gerade diesen Pixel. Diese Form einer diskreten Optimierung wird in Abschnitt 3.5.1.2 näher in ihrer Umsetzung beschrieben. Sie wird zu einem großen Teil durch die sogenannte *Pixelorder* repräsentiert.

Eine Bedingung ist die Optimierung der LED-Werte bzw. Duty-Cycles in Hardware in Echtzeit durchführen zu können. Dabei sollen auch die Hardwarekosten der Logik und des Speichers gering gehalten werden. Mit diesen Anforderungen kommt nur eine Pixelpipeline basierte Hardware-Architektur wie auch für das

Global-Dimming eingeführt, infrage. Aus dem Optimierungsproblem aus Abschnitt 2.5.3 geht hervor, dass für ein typisches 12,3 Zoll automotive ICD mehr als  $> 4 \cdot 10^6$  Ungleichungen erfüllt werden müssen. Die zeitliche Limitierung ist dabei die Bildwiederholrate. Zudem wird wie bereits beschrieben, die Information aus der Einflussmatrix benötigt, welche für dieses Beispiel mit  $720 \times 1920$  Pixel und  $N_L = 24$  (LED-Stränge) einen Speicherbedarf bei einer Auflösung von 12 Bit von 398,1312 MBit hätten. Diese riesige Datenmenge übersteigt den verfügbaren SRAM eines Mittelklasse-FPGAs um den Faktor 10 bis 20 und entspricht mehr als dem 10-fachen Speicherbedarfs eines Bildes dieser Auflösung bei einer Farbtiefe von 24 Bit. Um den Berechnungsaufwand der Ungleichungen sowie den Speicherbedarf zu reduzieren, wird das Eingangsbild sowie die LSF in der Auflösung reduziert. Das Verfahren wird in Abschnitt 3.5.1.1 genauer erläutert. Das Modul zur Reduktion wird als **Condenser** bezeichnet. Zusammen mit diesem Condenser, ergibt sich das SSC Schema wie in Abbildung 3.23 gezeigt. Das Prinzip der Pixelkompensation ist bereits aus den Grundlagen bekannt. Dieser Local-Dimming Algorithmus lässt sich somit in 3 Phasen unterteilen, wie in Abbildung 3.24 weiter unten dargestellt wird. In Rot wird dabei der Bilddatenfluss dargestellt, während die schwarzen Pfeile die LED-Wert-Bestimmung und Übertragung darstellen.

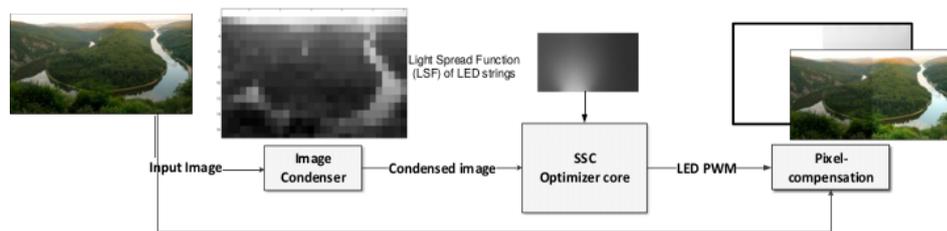


Abbildung 3.23.: Veranschaulichtes Schema des SSC Algorithmus nach SoA. Entnommen aus [Sch17].

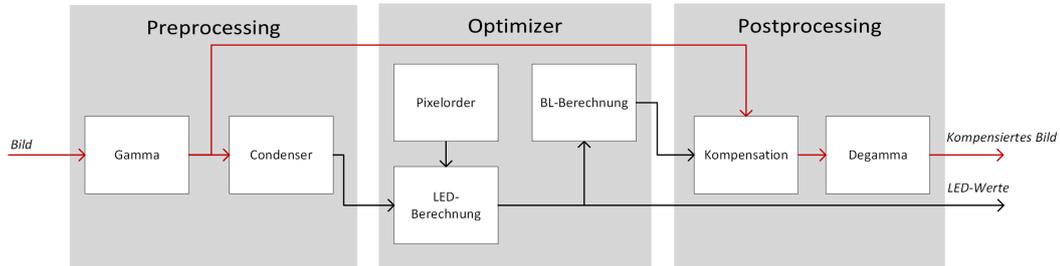


Abbildung 3.24.: Datenflussdiagramm des SSC Algorithmus nach SoA.

In der Phase des **Preprocessings** werden die Bilddaten zunächst in den physikalische linearen Bereich durch eine Gammafunktion transformiert. Basierend darauf werden die Pixel im Condenser zusammengefasst. Mithilfe der Pixelorder, welche eine Sortierung und eine Abarbeitungsreihenfolge der Pixel enthält, wird auch die LSF gespeichert. Mit den Condenserdaten wird im **Optimizer** schließlich die LED-Wertberechnung nach Abschnitt 3.5.1.2 durchgeführt. Wichtig zu erwähnen ist, dass nicht über die LEDs, sondern über die Condenserezellen iteriert wird. Zudem wird das auf der LED-Lösung basierende Hintergrundlicht hinter den Pixeln berechnet, sodass diese Information in der **Postprocessing** Phase für die Erhöhung der Transmission genutzt werden kann. Zum Schluss werden die Bilddaten wieder in den wahrnehmungslinearen Bereich mithilfe der Degamma Funktion überführt. In der Hardwareimplementierung, welche auf der Pixelpipeline basiert und keine Framebuffer nutzt, ist das Timing von großer Bedeutung, da jedes Pixel nur einmalig

### 3.5. Local-Dimming

durch die Pipelineregister geschoben wird. Die Berechnung der LED-Werte kann somit erst final erfolgen, sobald *alle* Pixel die Pipeline durchlaufen haben. Somit entsteht ein Versatz zwischen den Bilddaten und den LED-Werten um ein Frame. Bei einer FPS von 60 Hz stellt dieser Versatz jedoch kein Problem dar, da nacheinander folgende Bilder generell als ähnlich anzusehen sind. In Abbildung 3.25 wird der zeitliche Verlauf in der Implementierung verdeutlicht.

Die Logik wird hier in Grün repräsentiert, während die Read-Only Memorys (ROMs, Festwertspeicher) bläulich und RAMs rötlich hinterlegt sind. Die beschriebenen ROMs werden in typischerweise mit SRAM realisiert. Der Bilddatenfluss (Pixelpipeline) wird mit grünen Pfeilen visualisiert von RGB<sub>in</sub> nach RGB<sub>out</sub>. Dabei entsteht ein Bilddelay im Bereich von nur 40 Pixelclock Takten und liegt (abhängig von der Pixelclock) im unteren dreistelligen Nanosekunden Bereich und ist somit vernachlässigbar. Außerdem wird deutlich, dass der Optimizer erst nach dem jedes Pixel gesehen wurde starten kann um über alle Informationen (Condenserverte, siehe nächster Abschnitt) zu verfügen. Weiterhin ist erkennbar, dass die Pixelkompensation auf der LED-Lösung des vorangegangenen Bildes basiert. Das berechnete Backlight wird dabei im sogenannten (SP) Sample-Point-RAM gespeichert.

Die LED-Werte liegen nach der Berechnung in einem RAM oder Registern und werden meist über ein serielles Übertragungsverfahren wie beispielsweise Serial Peripheral Interface (SPI) zum LED-Treiber übertragen.

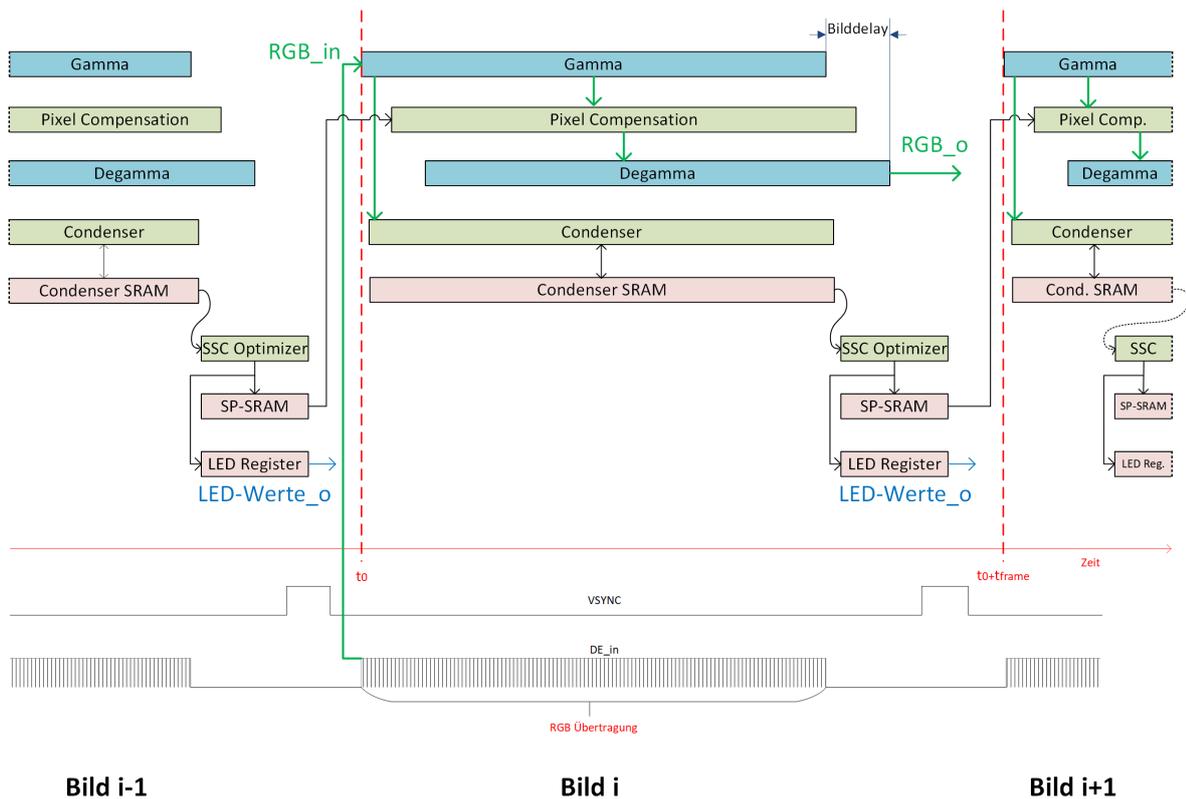


Abbildung 3.25.: Prinzipieller zeitlicher Ablauf des SSC Algorithmus nach SoA.

## 3.5.1.1. Preprocessing

Um die Komplexität der Optimierungsproblems (Formel (2.22)) zu reduzieren und auf die vollständige Speicherung von Frames (Framebuffer) verzichten zu können, wird der *Condenser* eingeführt. Dieser hat das Ziel, eine definierte Anzahl an Pixeln aus der direkten Nachbarschaft durch einen *repräsentativen* Wert zu beschreiben [Alb+10]. Abbildung 3.26 zeigt die Funktion des Condensers. Es werden hier  $m \times n$  Pixel durch einen einzigen Wert (hier  $P_C$ ) repräsentiert. Es wird also eine Funktion  $C : \mathbb{N}_0^{m \times n} \rightarrow \mathbb{N}_0$  realisiert. Da in der Hardwareimplementierung nur Bitfolgen zur Repräsentation einer Zahl genutzt werden und diese durch natürliche Zahlen dargestellt werden können, wird die Funktion auf  $\mathbb{N}_0$  definiert. Dabei kann  $C$  auf

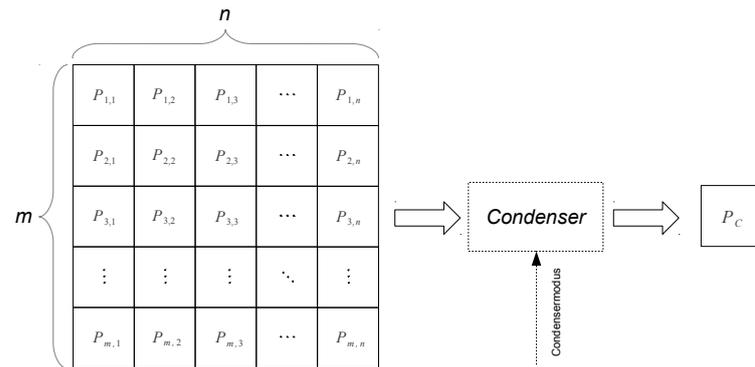


Abbildung 3.26.: Funktionsweise des Condensers mit Modi-Wahl. Entnommen aus [Sch13].

verschiedene Arten berechnet werden. An dieser Stelle sei der Maximalwertcondenser vorgestellt, da er für den in Kapitel 5 vorgestellten Algorithmus Verwendung findet. Seien die Werte der  $m \times n$  betrachteten Pixel durch den Vektor  $\mathbf{r} \in \mathbb{N}_0^{m \cdot n \times 1}$  repräsentiert, dann gilt für die Berechnung des Maximalwertcondensers

$$C_{\max}(\mathbf{r}) = \max_{r \in \mathbf{r}} \mathbf{r}$$

Mit diesem Condenser kann eine Clippingfreiheit garantiert werden, da die Pixel insgesamt durch ihre höchsten Werte repräsentiert werden. Die Werte  $\mathbf{r}$  sind hier bereits die maximalen Postgamma-Subpixel damit sichergestellt werden kann, dass für jeden Subpixel die geforderte Helligkeit erreicht werden kann. So wird der Lichtbeitrag für den größten Wert berechnet und deckt somit garantiert auch die kleineren Subpixelwerte eines Pixels ab.

Ein weiterer Condensermodus ist der sogenannte Mittelwertcondenser. Dieser erlaubt ein höheres Powersaving. Die Werte werden berechnet zu

$$C_{\text{mean}}(\mathbf{r}) = \frac{\sum_{i=0}^{m \cdot n - 1} \mathbf{r}(i)}{m \cdot n}$$

Somit stellen die Condenserwerte auch die Optimierungsziele dar und erlauben einen Trade-Off zwischen Powersaving und Clippingfreiheit. Es existieren auch weitere Modi wie den Automatik-Condenser [Sch17; Alb10], welcher den Mittelwertcondenser als untere Schranke betrachtet und eine Gewichtung  $f_{\text{auto}}$  zwischen dem Maximal- und Mittelwertcondenser abhängig von weiteren Parametern wählt. Es ergibt sich dann die

Funktion

$$C_{\text{auto}}(\mathbf{r}) = C_{\text{mean}}(\mathbf{r}) + f_{\text{auto}} \cdot (C_{\text{max}}(\mathbf{r}) - C_{\text{mean}}(\mathbf{r}))$$

mit  $f_{\text{auto}} \in [0; 1]$ . Im Allgemeinen Fall liefert ein  $f_{\text{auto}} = 0,5$  einen guten Kompromiss [XAJ16]. Für Consumer Geräte eignet sich diese Art des Condensierens gut, da die Bildinhalte, die im Rahmen eines Unterhaltungscontents mit einer „Kamera“ erzeugt werden, oft ein leichtes Clipping erlauben, ohne dass diese Artefakte bewusst wahrgenommen werden. Für den Einsatz im Automobil eignet sich aus Qualitätsgründen nur die Nutzung des Maximalwertcondensers. Für ein Display mit einer Full-HD Auflösung ( $1080 \times 1920$ ) werden typischerweise  $64 \times 64$  Pixel durch einen Condenserpixel repräsentiert. Aufgrund der speziell ausgewählten Lage der Condenserzellen (siehe Abschnitt 3.5.1.2) ergeben sich für diesen Fall 558 Condenserzellen, welche das gesamte Bild repräsentieren und die Zielfunktion der Optimierung darstellen. Diese Anzahl wird hier als  $N_C$  bezeichnet mit  $N_C^r$  Zeilen und  $N_C^c$  Condenserspalten. Somit gilt  $N_C = N_C^r \cdot N_C^c$ . Damit wird die Anzahl der Ungleichungen drastisch reduziert und entschärft das Problem der zeitlichen Limitierung der maximalen Berechnungszeit, welche durch die Bildwiederholfrequenz vorgegeben ist. Diese Reduktion kann durchgeführt werden da  $N_L \ll N_P$  angenommen werden kann. Die Änderungsrate der LSF innerhalb weniger benachbarter Pixel ist sehr gering, sodass eine Pixelgruppe durch einen einzigen Wert repräsentiert werden kann. Nach der Reduktion ist die Anzahl der LEDs immer noch deutlich geringer als die der Condenserzellen und gilt insbesondere für Edge-Lit. Diese extrahierten Daten werden in Hardware in einem SRAM gespeichert. In Abbildung 3.27 wird die Funktionalität des Condenser verdeutlicht. Es ist das originale Eingangsbild sowie dessen maximale Subpixel zu sehen. Darunter ist das Resultat des Mittelwert- und Maximalwertcondensers abgebildet. Es wird deutlich, dass bei dem Mittelwertcondenser die LED-Werte deutlich geringer berechnet werden als für den Maximalwertcondenser, da die Zielwerte geringer ausfallen. Insbesondere im unteren Bereich wird durch das fehlende Backlight ein Clipping in manchen Bildteilen induziert. Als Gegenmaßnahme kann durch eine geschickte Begrenzung der Kompensation das Clipping zwar reduziert werden [AX09], jedoch wird an diesen Stellen das Bild deutlich dunkler, als ohne Local-Dimming. Für den Consumer Bereich ist dies ein sehr guter Trade-Off, da hohe Einsparraten ermöglicht werden und diese Abdunklung bei natürlichen Bildern nicht negativ auffällt. Für den Automobileinsatz, insbesondere für den ICD, gilt es jedoch eine Abdunklung für Grauwerte  $> 0$  zu vermeiden, um die ursprüngliche Leuchtdichte möglichst gut zu treffen und den künstlichen Bildinhalt nicht lokal unterschiedlich abzudunkeln, da hiermit auch das HMI Design unerwünscht verändert werden würde. Dies begründet die weiter oben erwähnte Wahl für den Maximalwertcondenser im Automobil im weiteren Verlauf der Arbeit.

#### 3.5.1.2. SSC-Optimierer mit Umsetzung der LSF und Pixelorder

Im nächsten Schritt durchlaufen die Daten den *Optimizer*. Dieser ist für die Berechnung der LED-Werte unter Beachtung der Zielwerte, welche durch die Condenserwerte aus dem SRAM repräsentiert werden, zuständig. Hier wird das eingangs erläuterte lineare Optimierungsproblem in Hardware unter zeitlichen Constraints gelöst. Hierbei wird iterativ vorgegangen und sich der Lösung angenähert. Eine wichtige Besonderheit ist, dass ein LED-Wert niemals reduziert werden darf, sondern monoton steigen muss. Mit dieser Prämisse wird verhindert, dass in einer folgenden Iteration mehr *Clipping*, also eine größere Differenz zwischen der Soll- und Ist-Helligkeit einer Condenserzelle erzeugt wird. Dies führt dazu, dass der Algorithmus terminiert, da die Anzahl der Iterationen  $N_{\text{iter}}$  begrenzt ist. Somit kann das aktuell betrachtete Backlight hinter einem Pixel bzw. hinter einer Condenserzelle niemals kleiner werden, als in der Iteration zuvor. Zudem

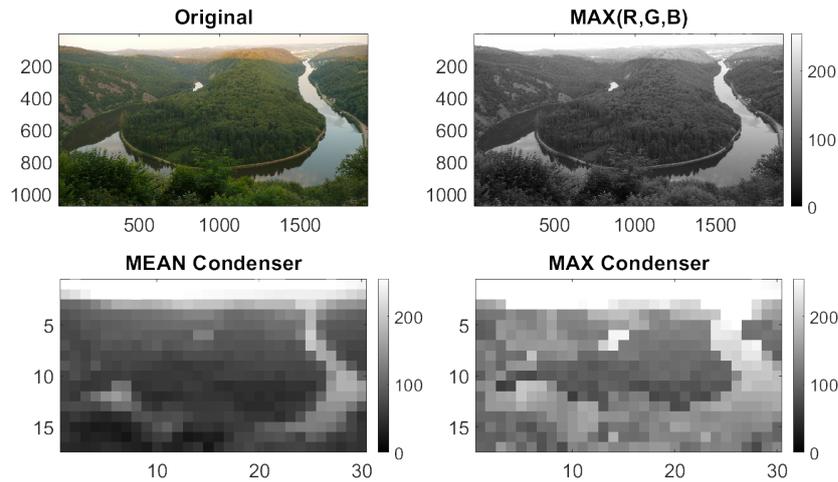


Abbildung 3.27.: Visualisierung des Condenserinhaltes für das *Saarschleifen* Bild für zwei Condensermodi, basierend auf den maximalen Subpixeln.

wird nicht von Anfang an der tatsächliche Zielwert (Condensierzelle  $c(i, j)$ ) abgedeckt, wobei die Matrix  $\mathbf{c} \in \mathbb{N}_0^{N_C^r \times N_C^c}$  die kondensierten Werte enthält, sondern ein verringerter Wert, welcher sukzessive bei jeder Iteration  $t \in \{0, \dots, N_{\text{iter}} - 1\}$  sich weiter dem tatsächlichen Condenserwert nähert. Das Backlight  $\check{b}_{i,j}$  berechnet sich mithilfe der Einflüsse wie in Formel (2.18) auf Seite 34 gezeigt.

Die Verringerung des Zielwertes wird laut [Sch17] durch eine Subtraktion von dem aktuellen Condenserwert realisiert. In dieser Arbeit wird jedoch diese Funktion zur Reduktion eines Condenserwertes verallgemeinert dargestellt mit  $\lambda(c(i, j), t)$ . Somit kann ein  $\Delta_{\check{L}}(i, j)$  berechnet, welches die Differenz des Soll-Wertes zur tatsächlichen Abdeckung in der aktuellen Iteration  $t$  quantifiziert wie in Formel (3.4) gezeigt.

$$\underbrace{\Delta_{\check{L}}(i, j)}_{\text{Differenz zum Soll-Wert}} = \underbrace{\lambda(c(i, j), t)}_{\text{Soll-Wert}} - \underbrace{\check{b}_{i,j}}_{\text{aktueller Lichtbeitrag}} \quad (3.4)$$

Dieser fehlende Beitrag muss folglich durch Erhöhung eines LED-Wertes ausgeglichen werden. Um die Summe der LED-Duty-Cycles zu minimieren, muss also diejenige LED erhöht werden, welche zu der betrachteten Condensierzelle  $(i, j)$  den höchsten Einfluss bzw. Beitrag liefert. Dies entspricht der *Haupt-LED*. Dadurch wird sichergestellt, dass tatsächlich der Duty-Cycle der in diesem Fall optimalen LED erhöht wird und somit die Duty-Cycles minimal bleiben. Würde eine LED erhöht, welche einen geringeren Einfluss ausübt, so muss der Duty-Cycle höher werden (siehe Formel (2.20), Seite 34), um die benötigte Helligkeit welche dem Zielwert entspricht zu erzeugen. Somit wäre die LED-Lösung jedoch nicht optimal. Sollte die Haupt-LED in der aktuellen Iteration bereits den Duty-Cycle von 100 % erreicht haben, so wird die LED mit dem zweithöchsten Einfluss erhöht usw.

Die in Formel (3.4) berechnete Differenz zum Soll-Wert wird zur Berechnung des LED-Wert-Inkrementes genutzt. Dies entspricht dem Wert, welcher dem Inkrement des aktuellen PWM-Wertes der LED  $l$  entspricht, um das fehlende Backlight an der Position  $(i, j)$  abdecken zu können. Es wird als  $\Delta_v(l)$  bezeichnet und wie

in Formel (3.5) berechnet.  $\check{L}_{(i,j),l}$  stellt den Einfluss der LED  $l$  in Condensierzelle  $(i, j)$  dar.

$$\begin{aligned} \Delta_{\check{L}}(i, j) &\stackrel{!}{=} \Delta_v(l) \cdot \check{L}_{(i,j),l} \\ \Leftrightarrow \Delta_v(l) &= \frac{\Delta_{\check{L}}(i, j)}{\check{L}_{(i,j),l}} \end{aligned} \quad (3.5)$$

Aus den beschriebenen Berechnungsvorschriften geht hervor, dass zur Berücksichtigung des Crosstalks, die LSF in Hardware gespeichert werden muss, sodass die Einflüsse jeder LED in jeder Condensierzelle bekannt sind. Zudem muss während jedem Schritt in der Lösungsfindung immer bekannt sein, welche Condensierzelle betrachtet werden muss, um eine optimale Lösung bezüglich der Zielfunktion zu erhalten. Dazu wird eine Datenstruktur eingeführt, welche als *Pixelorder* bezeichnet wird und in [Sch17] eingeführt wurde. Diese Datenstruktur speichert sowohl die Einflüsse als auch die Reihenfolge der Condensierzellen, welche nacheinander betrachtet werden müssen. So muss der Optimierer  $N_{\text{iter}}$  mal diese Datenstruktur vom ersten bis zum letzten Eintrag abarbeiten. Der Inhalt der Pixelorder ist so organisiert, dass jedem Schritt immer ein Zustand folgt, welcher den größten Beitrag zur optimalen Lösung zu liefern vermag. Dieses Verfahren entspricht einem Greedy-Algorithmus [Col+09; Kor06] zur Optimierung der LED-Werte.

Für eine Condensierzelle ergibt sich eine Datenmenge von 418 Bit bestehend aus 10 Bit für die Condensersadresse und weitere  $24 \cdot (5 + 12)$  bit. 24 steht für die LED-Anzahl, 5 Bit für die Adresse der LED und 12 Bit für den LSF-Wert. Der benötigte SRAM Speicher beträgt in diesem Fall 233,244 kBit bei  $N_C^r = 558$  und somit nur einen Bruchteil (2,5 ‰) des benötigten Speichers für die Speicherung der Einflussmatrix für jeden Pixel.

Wird die beschriebene Sortierung für ein Edge-Lit Gerät angewandt, so kann es durch die Einflüsse bedingt passieren, dass einige LEDs häufiger den maximalen Lichtbeitrag zu einem Pixel liefern als andere. Anhand eines Prototyps mit 16 LED-Strängen ergibt sich die Verteilung wie in Abbildung 3.28a dargestellt. Dies führt jedoch dazu, dass diese LEDs entsprechend häufiger während der Abarbeitung der Pixelorder betrachtet werden.

Um eine Art *Fairness* zu ermöglichen, wurde in [Sch17] ein weiteres Constraint für die Sortierung der Pixelorder eingeführt. Dieser Constraint sorgt dafür, dass eine Haupt-LED erst wieder als solche betrachtet wird, wenn alle übrigen LEDs in den vorherigen Schritten gewählt wurden. Dadurch weicht die Sortierung jedoch vom Optimum ab, sorgt aber für eine gleichmäßige Betrachtung der LEDs wie in Abbildung 3.28b anhand eines Beispiels abgebildet. Somit wird auch die Belastung im Betrieb gleichmäßiger verteilt. Für die spätere Kompensation der Pixel wird die berechnete Helligkeit ( $\check{b}_{i,j}$ ) hinter einem Pixel  $(i, j)$  benötigt. Da die Optimierung der LED-Werte auf  $N_C^r \times N_C^c$  Condensierzellen basiert und während der Optimierungsschritte diese benötigten Helligkeiten bekannt sind (vgl. Formeln (2.18) und (3.4)) kann diese Information direkt mit gespeichert werden. Somit ergeben sich für den klassischen Edge-Lit Fall und FHD-Auflösung  $N_{\text{SP}} = N_C = 558$  Stützstellen oder auch *Sample-Points* (SP) für die Speicherung des berechneten Backlights im SRAM.

#### 3.5.1.3. Postprocessing

In der letzten Phase des Local-Dimming Algorithmus müssen die Transmissionen der Pixel erhöht werden, um die durch das Dimming verdunkelten Bereiche in ursprünglicher Leuchtdichte anzeigen zu können. Das

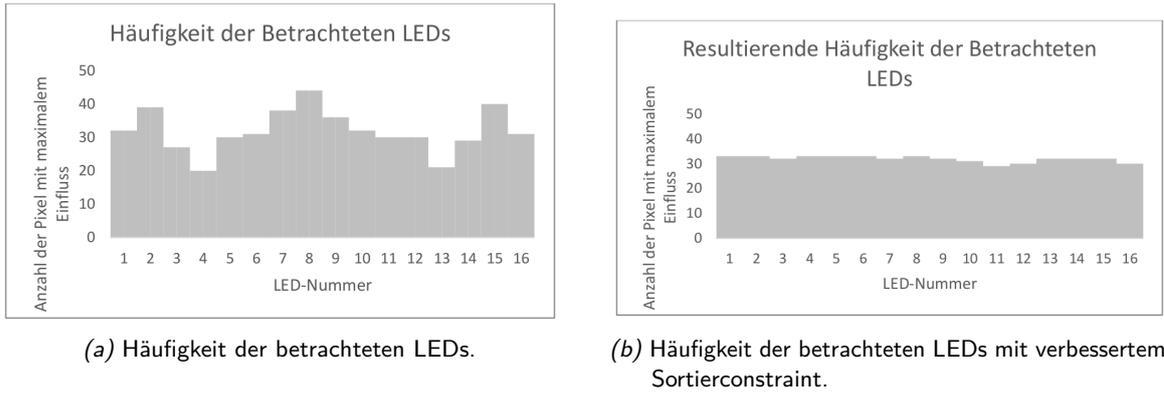


Abbildung 3.28.: Vergleich der Häufigkeiten der Betrachtung einzelner LEDs im Ablauf des SSC-Optimierers einer Edge-Lit BLU, basierend auf Änderung der Pixelordersortierung. Entnommen aus [Sch17, S. 31 f.].

Konzept wurde bereits in Abschnitt 2.5.4 eingeführt und wird wie folgt für jeden Punkt berechnet.

$$r_{i,j}^{\text{comp}} = \min \left( \frac{r_{i,j}}{\check{b}_{i,j}}, t_{\text{max}} \right) \quad (3.6)$$

Hierbei stellt  $t_{\text{max}}$  den maximal möglichen Transmissionswert dar und ist in der Regel 100 %. Falls  $r_{i,j}^{\text{comp}} > t_{\text{max}}$  rechnerisch ergeben würde, so läuft der (Sub-) Pixel ins Clipping.

Um nun zu jedem Pixel den Kompensationsfaktor  $(\check{b}_{i,j})^{-1}$  zu bestimmen, muss in jedem Punkt das resultierende Backlight bekannt sein. Als Skalarfeld wird also  $\check{\mathbf{B}}$  benötigt. Da jedoch die Datenmenge zu groß wäre um für jedes Pixel das entsprechende Backlight  $\check{b}_{i,j}$  zu speichern, wird es basierend auf den bereits zur Optimierungszeit berechneten Sample-Points bilinear [Pre+07] interpoliert. Somit berechnet sich der aktuelle Backlightwert zwischen vier Sample-Point Zentren als Interpolation (`interp`), welche über die Distanz des betrachteten Pixels zu den vier SPs horizontal und vertikal gewichtet wird, mit

$$\check{b}_{i,j} = \text{interp} \left( p_{i,j}^{\text{SP}}, \text{SP}_{(y,x)}, \text{SP}_{(y+1,x)}, \text{SP}_{(y,x+1)}, \text{SP}_{(y+1,x+1)} \right) \quad (3.7)$$

$p_{i,j}^{\text{SP}}$  stellt hierbei die Lage eines Pixels zwischen den SPs in Punkt  $(i, j) \in \{0, \dots, 63\} \times \{0, \dots, 63\}$  dar. Die Berechnung der Backlights und des resultierenden Kompensationsfaktors wird hierbei in Echtzeit auf der Pixelpipeline berechnet. Nach der Kompensation folgt wie beschrieben die Adaption der physikalischen Daten in den wahrnehmungslinaren Bereich und die Übertragung der Bilddaten an das Panel. Die Übertragung selbst wird auf dieser letzten Übertragungsstrecke oft mit einem auf dem LVDS Physikal Layer basierendem Protokoll durchgeführt (siehe Abschnitt 2.6). Die Adaption der Bilddaten erfolgt mithilfe der OETF und wird mit der inversen Gammakurve realisiert (*Degamma*). In [Sch17] wurde aufgrund des großen Speicherbedarfs, welcher für die Speicherung der Degamma LUTs benötigt wird, eine abschnittsweise Realisierung der Degamma-Kurven implementiert. An dieser Stelle wird auf die genannte Arbeit von Schäfer verwiesen.

Ein Beispiel der Kompensation ist in Abbildung 2.27 auf Seite 35 zu finden. Aufgrund der typischen HMI Strukturen wie Kreise, die den Tachometer repräsentieren sowie die steilen Übergänge, die einen hohen Grauwertkontrast definieren, hat sich in der Fachwelt die Erkenntnis durchgesetzt, dass hochauflösende Matrix-BLUs erforderlich sind, damit ein Local-Dimming System gute Ergebnisse liefern kann. Im nächsten Abschnitt wird speziell auf das Local-Dimming mit Matrix-Backlight eingegangen.

### 3.5.2. Direct-Lit Dimming und Stand der Technik

In diesem Abschnitt wird die Funktionsweise des gängigen Local-Dimming Ansatzes für Direct-Lit BLUs erläutert. Der Bedarf eines Local-Dimming Systems zur Verbesserung der visuellen Qualität sowie der Leistungsaufnahme eines LCDs im Consumer-Bereich belegen beispielsweise die Arbeiten von *Samsung* [Che+06; Yeo+08] und *AUO* [Lin+07] zu diesem Thema. Insbesondere standen LC-Fernseher im Fokus. Klassischerweise wird dabei das sogenannten *Downsampling* Verfahren verwendet wie im Folgenden vorgestellt wird.

#### Downsampling Ansatz

Ein tendenziell in Hardware kostengünstig zu realisierender Ansatz des SoA wird über die Methode des Downsamplings des Bildinhaltes von  $N_P^r \times N_P^c$  auf die Auflösung des Matrix Backlights  $N_L^r \times N_L^c$  realisiert. Dabei repräsentiert der jeweils maximale Pixel eine gesamte Pixel-Gruppe mit der Auflösung  $\frac{N_P^r}{N_L^r} \times \frac{N_P^c}{N_L^c}$ . Bleibt bei der Division ein Rest übrig, so müssen zusätzliche Pixelreihen oder Spalten *virtuell* erweitert werden, um ganzzahlige Ergebnisse zu erhalten. Zur Vereinfachung wird angenommen, dass es sich hier bereits um Postgamma-Werte handelt, welche sich proportional zur Lichtintensität verhalten. Somit kann jeder LED die Intensität bzw. LED-Wert des entsprechenden repräsentativen Wertes aus dem Downsampling Prozess zugeordnet werden. Dieses Verfahren wird in Abbildung 3.29 zur Veranschaulichung dargestellt. Der rot markierte Bildbereich wird dabei auf die darunterliegende LED abgebildet und stellt den LED Duty-Cycle dar. In [Shi+06] wurde solch ein blockweiser Ansatz prototypisch vorgestellt.

Der Kompensationsfaktor zur Erhöhung der Pixeltransmission nach Formel (2.23) (Seite 35) entspricht im trivialen Ansatz gerade dem Kehrwert der Lichtintensität und gilt für die ganze Pixelgruppe. In diesem Fall kommt es dadurch zu Artefakten wie beispielsweise sichtbare Kanten zwischen den Kavitäts Grenzen. Um dies zu verbessern, wurde in [Zha+09] eine Normalverteilung angenommen, um die Lichtintensität in Abhängigkeit von einem LED-Zentrum zu modellieren und somit die Pixelkompensation zu verbessern (siehe auch Abschnitt 4.3.1). Jedoch fehlt bei dieser Methode die Beachtung des Einflusses aller LEDs und kann somit nicht optimal sein. Zudem kann mit einer einzelnen LED niemals die ursprüngliche Leuchtdichte in einem weißen Pixel erreicht werden.

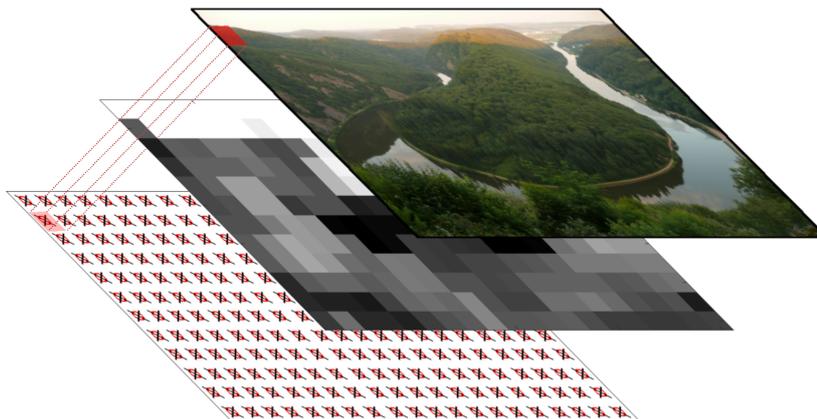


Abbildung 3.29.: SoA Ansatz durch *Downsampling* des Bildinhalts.

Ein weitaus komplexerer Ansatz wurde bereits in [See+04] vorgestellt. Zwar wird hier auch das Bild auf

die Auflösung der LED Matrix reduziert und das Maximum der Pixelgruppe als Zielwert angenommen, jedoch wird mithilfe der Point-Spread-Function (PSF), welche ebenfalls mit einer Normalverteilung angenähert wurde, die LED-Duty-Cycles zurückzurechnen. In Abbildung 3.30 wird dieses Verfahren abgebildet. Eine Faltung [Bro+08, S. 768] der LED-Intensitäten mit der PSF würde also für jeden Punkt eine LED-Intensität liefern. In der Abbildung werden als Eingangsgröße die Intensitätswerten  $I$  genommen (1).

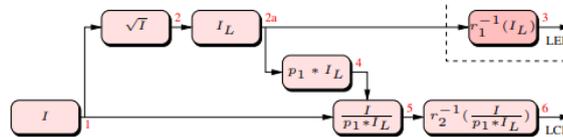


Abbildung 3.30.: Algorithmus für die LED-Wert-Berechnung nach von Seetzen u. a. Entnommen aus [See+04].

Für alle LEDs wird bei (2a) die Ziel-Intensitäten zu bestimmen ( $I_L$ ). An dieser Stelle wird der Bildinhalt auf die LED-Auflösung reduziert. Die einzelnen LED-Werte berechnen sich nun durch die Dekonvolution mit Kenntnis der PSF  $p_1$ . Dieser Vorgang ist jedoch für eine Echtzeit-Prozessierung völlig ungeeignet und numerisch instabil [See+04] und muss weiter vereinfacht werden, wodurch Fehler eingeführt werden. Nach der Lösung werden die LED-Werte mit einer Korrektur mit Kenntnis der Antwortfunktion  $r_1$  der LEDs an die LED-Treiber geschickt.

Die Berechnung des Backlights wird durch die Faltung (4) realisiert und das Bild kompensiert (5) und wird schließlich an das LC-Panel weitergeschickt (6).  $r_2$  entspricht hier der Antwortfunktion des Panels.

Ein großer Nachteil der SoA Methoden neben der Vernachlässigung des Crosstalks und grober Reduktion des Bildinhaltes auf die LED-Matrix Auflösung, ist die Notwendigkeit der Nutzung von Framebuffern. Diese induzieren eine Verzögerung des Bildes und erzeugen entsprechende Kosten bei einer Realisierung in Hardware. Eine Echtzeitprozessierung ist in diesen Fällen nicht möglich. Zudem stellt die bei einigen Methoden notwendige Entfaltungsoperation eine große Hürde dar. Diese Misstände können jedoch mit dem SSC Algorithmus überwunden werden, wie bereits für die Edge-Lit BLUs weiter oben gezeigt wurde (Abschnitt 3.5.1), das das SSC Local-Dimming Modell (Formel (2.22)) allgemein formuliert und nicht auf Edge-Lit eingengt ist.

### Problemstellung: SSC für Matrix-BLUs

Ein wesentlicher Unterschied zwischen dem SSC und den übrigen Ansätzen ist die Prozessierungsrichtung. Vor der Entwicklung des SSC war es üblich vom Bild herunter auf die LEDs zu rechnen („vom Pixel zur LED“), während der SSC Algorithmus den physikalischen Weg „von der LED zum Pixel“ nimmt, wie in Abbildung 3.31 dargestellt. Die Funktionsweise wurde bereits in Abschnitt 3.5.1 für Edge-Lit BLUs beschrieben.

Da der SSC im Allgemeinen unabhängig von der LED-Anzahl und Positionierung das Local-Dimming Minimierungsproblem (Seite 34) löst, gelten die beschriebenen Methoden im Prinzip auch für Direct-Lit BLUs. Da jedoch der Anspruch besteht, die entworfenen Algorithmen auch in Hardware umzusetzen, treten bei steigenden LED-Zahlen recht schnell neue Herausforderungen auf. Die Anforderungen sind unter anderem eine echtzeitfähige Realisierung ohne Verwendung von Framebuffern und gerade im Automobilbereich auch unter Minimierung der Hardwarekosten. Dabei darf jedoch die visuelle Qualität nicht degradiert

### 3.5. Local-Dimming

werden. In Abbildung 3.32 ist das Schema für die Direct-Lit SSC Umsetzung visualisiert und ist analog zu Abbildung 3.23 (Seite 62). In Abschnitt 3.5.1.2 wurden die Methoden zur Umsetzung der LSF und

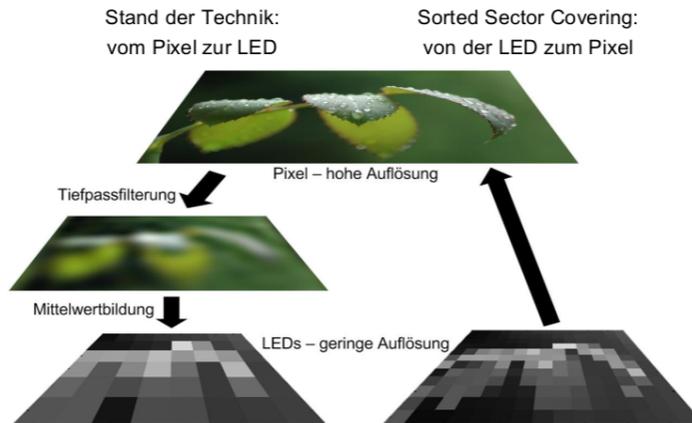


Abbildung 3.31.: Vergleich der Prozessierungsrichtung der klassischen Local-Dimming Ansätzen mit dem SSC. Entnommen aus [Alb10].

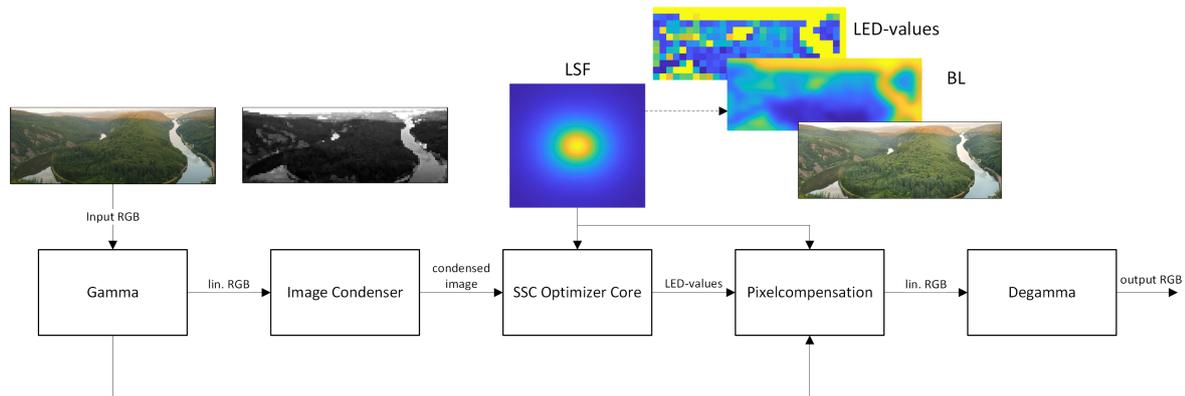


Abbildung 3.32.: Schema und Visualisierung der Processing Prozedur für Direct-Lit.

Pixelorder für Edge-Lit Geräte eingeführt. Wird nun solch eine BLU durch eine Direct-Lit Variante ersetzt, so ändert sich zunächst an der Prozessierung nichts. Bei gleichbleibender LED-Anzahl ist der Speicherbedarf exakt identisch ( $N_L = 24$ ). Steigt jedoch die Anzahl der LEDs, so steigt die Komplexität des SoA SSC Algorithmus überproportional. Die Breite der Pixelorderwörter erweitert werden auf die entsprechende LED Anzahl, welche durch die Einflussmatrix beschrieben werden. Anstelle einer Wortbreite von 418 Bit erweitert sich diese bei einer LED-Anzahl von 240, welche tatsächlich Anwendung findet, auf 4810 Bit. Insgesamt unter Annahme von identischen Condensierzellen würde hier ein Speicher von 2,68398 MBit benötigt werden, welcher die Kosten in inakzeptable Höhen treiben würde.

Hinzu kommt die Notwendigkeit die Anzahl der Condensierzellen zu erhöhen, also die Anzahl der Pixel pro Condensierzelle aufgrund der HMI-Inhalte zu verringern. Eine weitere Ursache ist die steilere LSF, da die BLU viel mehr LEDs beinhaltet. Wird die Auflösung zu grob, so gehen wichtige Informationen über den Einfluss verloren. Ein weiterer Grund ist, dass die Condensierzellen nach SoA bei 240 LEDs Matrix-Backlight mehrere LEDs abdecken und somit den Vorteil der höheren Lokalität verringert wäre. Die Lösung wäre

suboptimal. Das System würde hier die gewünschte Performance nicht erreichen können. Somit würde auch die Anzahl der Wörter in der Pixelorder ebenfalls steigen und den Speicherbedarf abermals erhöhen. Eine weitere Folge ist die Erhöhung der benötigten Zeit für den Optimierungsprozess, da die Pixelorder iterativ abgearbeitet wird. Somit wird pro Condensierzelle mehr Zeit benötigt, da mehr Einflüsse betrachtet werden müssen, sowie durch die gestiegene Condensierzellenanzahl auch die Länge der Pixelorder. Das zeitliche Limit, welche über die reziproke Bildwiederholfrequenz definiert ist, wird hier sehr schnell erreicht. Der gesamte Hardwarebedarf steigt stark an. An dieser Stelle wird auf Abschnitt 5.1 (ab Seite 129) verwiesen. Dort wird konkreter auf diese Thematik eingegangen.

### 3.5.3. Ansteuerschemata für eine hoch aufgelöste LED-Matrix

In diesem Abschnitt werden einige geläufigen Ansteuerschemata für die Direct-Lit BLUs vorgestellt. Je nach einzeln ansteuerbaren Dimming-Zonen wird zudem die Verdrahtung komplexer. Getrieben wird dabei typischerweise mit PWM bei konstantem Strom.

#### Direct-Drive-Ansteuerung

Diese Methode wird auch als *Single-Channel-Ansteuerung* bezeichnet. Hierbei wird jede LED bzw. als eine Einheit verschaltete Dimming-Zone bestehend aus mehreren LEDs, direkt mit einem LED-Treiber (Strom-Senke) bzw. einem Kanal eines LED-Treibers verbunden [Tex18]. Es existieren LED-Treiber ICs mit typischerweise zwischen 8 und 48 Kanäle. In Abbildung 3.33 ist das Schema für eine 4 × 4 LED-Matrix abgebildet. Es werden hierfür beispielhaft 2 Treiber mit jeweils 8 Kanälen genutzt, welche über eine serielle Daisy-Chain verbunden sind. Oft wird hierzu ein Serial Peripheral Interface (SPI) genutzt. Die LED-Werte bzw. die Duty-Cycles werden von einem Local-Dimming Prozessor berechnet.

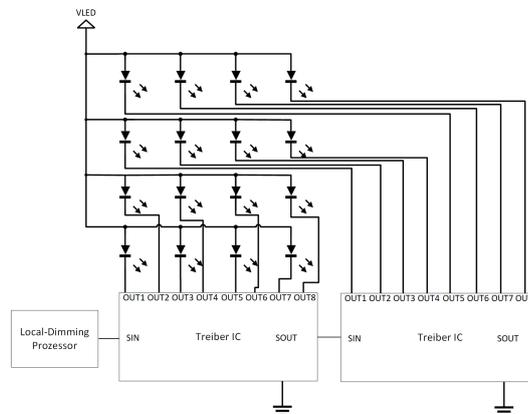


Abbildung 3.33.: Schematische Darstellung der Direct-Drive-Ansteuerung von 16 LEDs.

#### Passiv Matrix

Um ICs und somit auch die Kosten einzusparen, kann ein Multiplexing Ansatz gewählt werden. Dabei werden nicht alle LEDs gleichzeitig mit einer jeweils eigenen Stromsenke betrieben, sondern in Gruppen, welche zeitlich sequenziell angesteuert werden. Ein Beispiel sei ein 4:1 Zeilenmultiplex in Abbildung 3.34 gegeben. Die zeitliche Reihenfolge wird verdeutlicht. Es werden 25 % aller LEDs gleichzeitig angesteuert. Diese haben

### 3.5. Local-Dimming

zum Ansteuerzeitpunkt jeweils eine eigene Verbindung zu einer Stromsenke. Da die LEDs in diesem Fall nur  $\frac{1}{4}$  der Leuchtzeit haben, muss der Strom um den Faktor 4 erhöht werden um dieselbe Helligkeit zu erzeugen. Es kommt hinzu, dass die Stromeffizienz kleiner und die Vorwärtsspannung größer wird. Der Leistungsumsatz steigt. Somit ist der genannte Faktor 4 tatsächlich noch höher und die thermische Zeitkonstante wird größer.

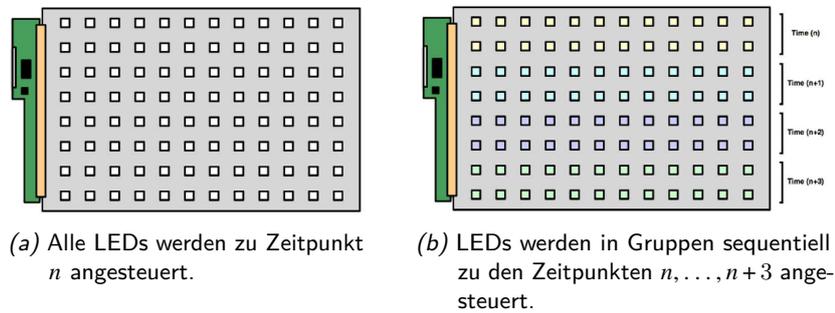


Abbildung 3.34.: Vergleich der zeitlichen Ansteuerung im Multiplexverfahren. Bilder entnommen aus [Cum19].

Das Schema so einer Multiplex-Schaltung ist in Abbildung 3.35 anhand eines einfachen Beispiels eines 2:1 Zeilenmultiplex bei einer  $4 \times 4$  LED-Anordnung abgebildet. In Rot ist dabei die *gemeinsame* Kathode zweier LEDs visualisiert. Diese sind an *einem* Treiberkanal angeschlossen. Durch die Schalter  $S_1$  und  $S_2$  wird jedoch die Anode zeitlich separiert. Diese Methode führt zu dem Vorteil, dass weniger LED-Treiber benötigt werden und das Routing der LED-Kathoden innerhalb der BLU einfacher wird. Jedoch verursachen  $S_1$  und  $S_2$  zusätzliche Verlustleistung (Schaltverluste). Die Effizienz nimmt ebenso etwas ab. Insgesamt muss auch auf das Timing geachtet werden. Die Logik für dieses Treiberschema wird also komplexer. Neben den Kathoden (durch die Treiberkanäle geschaltet), müssen für diesen Fall auch die Anoden berücksichtigt werden. Bei der Auswahl ist auf die maximalen Ströme zu achten, da sie beim Zeilenmultiplex entsprechend hoch sein müssen. Für sehr hochauflösende BLUs können sogenannte Mini-LEDs (siehe Abschnitt 7.2) genutzt werden.

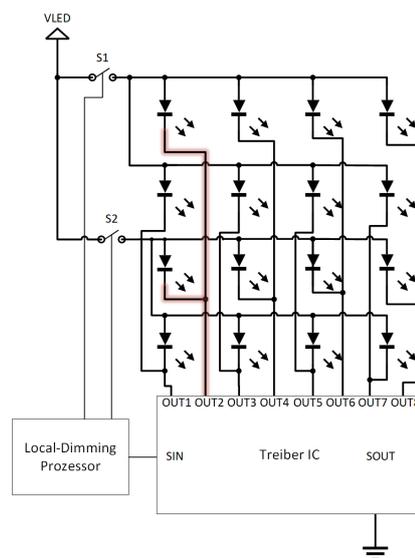


Abbildung 3.35.: Schematische Darstellung der Direct-Drive Methode unter Nutzung von Zeilenmultiplex.

Dabei wird oft auf ein *Aktiv-Matrix* Treiberschema zurückgegriffen, welche trotz hoher LED-Zeilen mit der einfachen Stromamplitude arbeiten kann. Jedoch verursacht das zusätzliche Aktiv-Matrix-Substrat erhebliche Kosten.

### Erhöhung der Effizienz

Bei einem LED-Treiber wird ein Spannungsabfall  $U_d$  benötigt der für eine Verlustleistung durch den Diodenstrom ( $I_d$ ) sorgt, welcher durch die Senke fließt. Zum Zeitpunkt der Ausarbeitung dieser Monografie, ist die Nutzung von 3V LEDs in den BLUs üblich. Wird stattdessen eine 6V LED genutzt, welche aus zwei seriellen 3V LEDs in einem Chip realisiert sein kann, so kann der Strom für die in Summe gleiche Photonenemission im Optimalfall halbiert werden. Sei im Allgemeinen der Lichtausbeutefaktor als  $f$  bezeichnet (hier  $\approx 2$ ). Dann kann die Verlustleistung im Treiber von  $P_{3V} = U_d \cdot I_d$  auf  $P_{6V} = U_d \cdot \frac{I_d}{f}$ , also gerade um den Faktor  $f$  reduziert werden.

Da im Automobil die Versorgungsspannung des Bordnetzes auf unter 9V temporär Abfallen kann, wird empfohlen die BLUs mit 6V LEDs auszustatten, um die Verlustleistung zu reduzieren. Eine weitere Erhöhung der Vorwärtsspannung ist für den Automobilbereich nicht zu empfehlen, da ansonsten wegen der möglichen Spannungseinbrüche nach unten, Buck-Boost-Converter verwendet werden müssten. Diese sind jedoch nicht nur teurer als reine Buck-Converter, sondern auch weniger effizient. Daher wurde eine solche LED dem Projektpartner *Nichia* empfohlen, was auch umgesetzt wurde.

## 3.6. Schlussfolgerungen

Zusammengefasst wird festgestellt, dass automotive Anwendungen besonders hohe Herausforderungen an Displays und auch an Dimmingverfahren darstellen. Bei typischen künstlichen HMI Inhalten kommen häufig sehr kontrastreiche Übergänge vor, welche aufgrund der Kanten auffallen. Dazu kommen hohe Anforderungen an die Peak-Leuchtdichte und eine geringe Fehlertoleranz aufgrund des künstlichen Inhalts. Eine hohe Lebensdauer und Zuverlässigkeit wird ebenfalls erwartet und muss bei hohen Betriebstemperaturen erreicht werden.

Global-Dimming wurde durch die Bildanalyse erweitert. Für Consumer-Geräte bzw. für natürliche Bilder ist er gut geeignet und bringt begrenzte Vorteile für die automotive Anwendungen.

Edge-Lit mit dem SoA SSC ist gut geeignet für TVs, genügt jedoch nicht der automotive Anwendung. Die bisherigen Methoden und Implementierung würden bei hohen LED-Zahlen (LED-Auflösungen) wirtschaftlich untragbare Kosten verursachen und den Echtzeitbetrieb (Video) nicht schaffen. Somit sind neue Konzepte und Architekturen notwendig.

Für automotive Anwendungen ist eine hoch-auflösende Matrix BLU mit Local-Dimming zielführend. Die Herausforderungen ist weiterhin ein effizienter Algorithmus in Kombination mit einem gut geeigneten BLU-Design. Letzteres ist nach dem Stand der Veröffentlichungen bisher kaum erforscht (Impact von BLU auf Local-Dimming). In dieser Arbeit wird daran gearbeitet, einen guten Kompromiss zwischen Local-Dimming Ergebnis und BLU-Kosten zu erreichen. Dabei wird das Zusammenspiel bzw. der Impact von der BLU mit ihren LSFs und dem Local-Dimming Algorithmus untersucht. Im nächsten Kapitel wird auf die hochauflösenden BLUs eingegangen. Dabei werden auch die Unterschiede zu Edge-Lit BLUs aufgezeigt und einige Aspekte des BLU Design-Ziels vorgestellt. In Kapitel 5 und 6 werden neue Ansätze und Methoden eingeführt, um die Effizienz und Bildqualität eines LCDs deutlich zu steigern bei gleichzeitiger Erfüllung der automotiven Anforderungen.

## 4. Hochauflösende Backlight-Units

Wie in Kapitel 3 bereits beschrieben wurde, sind die Anforderungen an ein Display in der automobilen Anwendung deutlich höher als an Displays aus dem Bereich der Unterhaltungselektronik. Insbesondere gilt dies für das ICD. Die Bildqualität muss hoch und die Ablesbarkeit garantiert werden (die Bilddaten bzw. Zielwerte müssen wiedergegeben werden können), auch unter Sonneneinstrahlung. Für letztgenannten Fall sind sehr hohe Leuchtdichten erforderlich. Gleichzeitig darf die Stromaufnahme ein Maximum nicht überschreiten, da ansonsten die Verlustleistung in Form von Wärme nicht mehr abgeführt werden kann. Der enge Bauraum für das ICD macht ein durchdachtes Design des Display-Moduls unumgänglich. Zudem muss das Display im Nachtbetrieb hochwertig wirken und einen guten Schwarzwert liefern. Insbesondere für die Premium-OEMs stehen Bildqualität und Wertigkeit im Vordergrund.

Um diesen Ansprüchen zu genügen, wird für Direct-Lit BLUs versucht die Anzahl der LEDs (genauer genommen die einzeln ansteuerbaren LED-Stränge bzw. Dimming Zonen) weiter zu steigern. Dies wird unter anderem durch sinkende LED Preise ermöglicht. Mit einem adäquaten Local-Dimming Algorithmus können die beschriebenen Anforderungen erfüllt werden.

Eine höhere Anzahl von Dimming Zonen wird generell vom Betrachter favorisiert [Swi+06]. Insbesondere bei niedrigem nativen Panel Kontrast ist eine höhere Zonenanzahl zur Verbesserung der visuellen Qualität erforderlich [HSX16a; Hua+19; Tan+18].

Im Bereich der Unterhaltungselektronik besitzt beispielsweise das TV-Gerät *Hisense 65XT910* 240 einzeln ansteuerbare LEDs, deren LSF in Abschnitt 4.1.2 untersucht wird. In [Xia+19] wird ein TV-Gerät mit 5184 Zonen vorgestellt. Im Größenbereich von Smartphones werden BLUs mit 288 bis zu 10368 Zonen bestehend aus Mini-LEDs beschrieben [Den+18; Tan+18]. In Abschnitt 7.2 wird näher auf den Aufbau solcher BLUs eingegangen.

Im Automobilbereich wurde von *Tianma Microelectronics Co., Ltd.* 2018 ein 12,3 Zoll ICD mit 96 Dimming Zonen gezeigt [Mek18] und verdeutlicht den Bedarf der OEMs. Die Firma *Texas Instruments Incorporated* (kurz: *TI*) stellt ein Local-Dimming Referenzdesign basierend auf 144 Dimming Zonen zur Verfügung [Tex18] und unterstreicht damit die Wichtigkeit von Direct-Lit BLUs im Automobil und sieht einen wichtigen Absatzmarkt für ihre LED-Treiber wie beispielsweise dem *TLC5955* welcher unter anderem in den verwendeten Prototypen Verwendung finden. Wei-Fu u. a. verwendeten BLUs mit 128 und 384 Zonen mit einer Local-Dimming Software-Implementierung für ein ICD [Wei+18]. Eine eigene komplette BLU wird von der *Nichia Corporation* unter dem Produktnamen *Nichia Light Cluster Type J* vermarktet und besitzt 240 Dimming Zonen [MNS19]. Im Rahmen einer Zusammenarbeit wurde dieses Cluster als Prototyp verwendet. *Osram Opto Semiconductors* ist der zweite führende LED-Hersteller hinter *Nichia* und auch forscht an Direct-Lit BLUs sowie LEDs [Kru+17]. In Kooperation mit der *Continental Automotive GmbH* entstanden Local-Dimming Systeme mit 133, 240 und 319 Dimming Zonen und stießen auf großes Interesse seitens der OEMs. Die konkreten Prototypen und Hardwaredesign werden in Abschnitt 5.6 vorgestellt. Abbildung 4.1 zeigt die BLU mit 319 Dimming Zonen bzw. LEDs aus der genannten Kooperation. Im Bild sind ebenfalls die Anschlüsse zur LED-Steuerung sichtbar. Für gute Resultate bezüglich der wahrgenommenen Wertigkeit und hohe Reduktion des Leistungsbedarfs zu erreichen, ist das BLU-Design sehr wichtig.

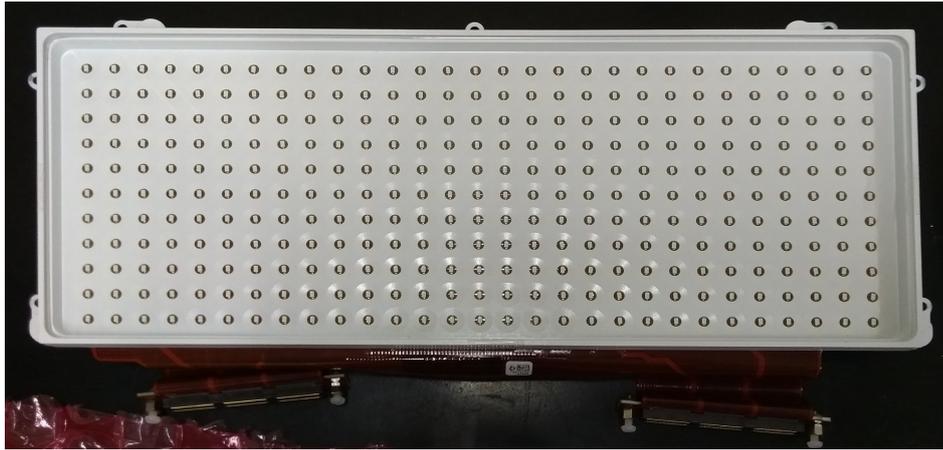


Abbildung 4.1.: BLU Prototyp mit 319 LEDs.

### Allgemeine Aspekte des BLU-Designs

Die Abbildung 4.2 stellt die in dieser Arbeit herausgearbeiteten Zusammenhänge der BLU Design-Parameter und Local-Dimming dar. Die gestrichelten Linien zeigen an, dass diese Beziehungen durch die Einführung von Local-Dimming entstehen. So kann aufgrund des in Abschnitt 2.4 beschriebenen *Light Leakage* bei einem LCD das sog. *Halo*-Artefakt entstehen, sobald ein Local-Dimming Algorithmus eingesetzt wird. Die Ursache dafür ist die unvollständige Blockade des Lichtes bei einem Grauwert von 0. Darauf wird in Abschnitt 6.2 näher eingegangen. Die bereits aus Abschnitt 2.5.2 bekannte LSF liefert aus Local-Dimming Sicht, die

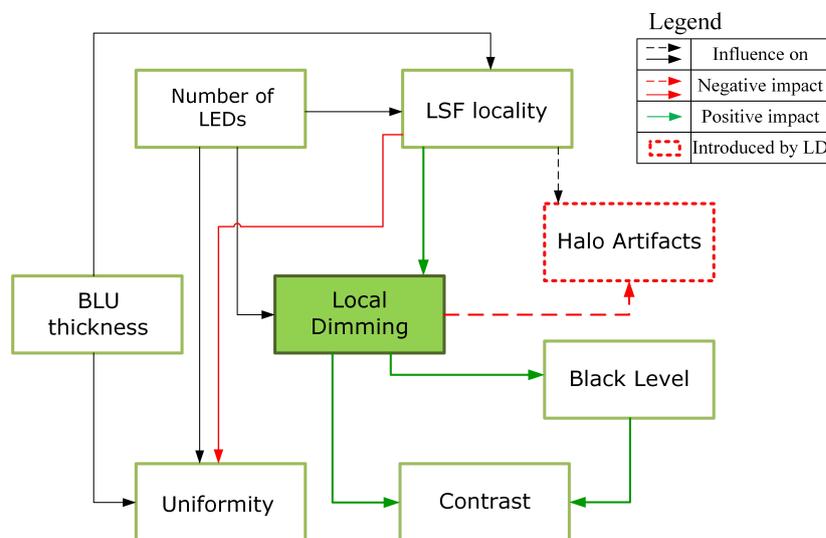


Abbildung 4.2.: Abhängigkeiten und Beziehungen der BLU Design-Parameter [Sch+18].

wichtigsten Informationen bezüglich des Potenzials einer BLU. So können beispielsweise zu steile LSF in Kombination mit unzureichenden Dimming-Algorithmen das Halo Artefakt verstärken. In Abschnitt 4.1 wird die damit zusammenhängende Lokalität und Einfluss der LSFs untersucht.

Aufgrund des knappen Bauraumes muss die gesamte Moduldicke inklusive BLU entsprechend dünn entwickelt

werden. Zusätzlich wird aus unternehmerischer Sicht seitens der OEMs versucht die Modulpreise ihrer Zulieferer möglichst gering zu halten. Dabei wird an der LED-Anzahl selbst, an den LED-Treibern und auch an der benötigten Chipfläche (oder FPGA-Größe) des Algorithmus gespart und kann folglich zur Leistungsreduzierung und Kompromissen führen. Dies rechtfertigt die in Kapitel 5 beschriebenen Optimierungen des Local-Dimming Algorithmus für eine effiziente Hardwareumsetzung unter Berücksichtigung des Kostendrucks. Trotz dieser Restriktionen ist auch wie weiter oben beschrieben, die Tendenz der fortan steigenden Dimming Zonen gegeben. Die Ursache liegt an dem hohen Potenzial der Reduktion der Leistungsaufnahme und an höheren Kontrast sowie an verbessertem Schwarzwert durch Local-Dimming.

## 4.1. Untersuchung des Einflusses auf die Pixel und Einfluss auf Local-Dimming

In den folgenden Kapiteln wird der Begriff *LED-Anzahl* anstelle von *Anzahl der Dimming Zonen* für den BLU Kontext verwendet. In diesem Fall bezieht sich der Begriff *LED* in einer Formel stellvertretend auf einen separat steuerbaren LED-Strang oder einer Dimming Zone.

Zunächst wird festgelegt, wie die LEDs indiziert werden. Dazu ist in Abbildung 4.3 eine beispielhafte LED-Anordnung in einer Direct-Lit BLU dargestellt. Die LEDs können entweder in linearisierter Form indiziert werden oder mithilfe der Angabe der entsprechenden LED-Zeile und -Spalte. Sei die Menge aller vorhandenen LEDs mit  $\mathcal{L}$  gegeben. Dann gilt für die Gesamtanzahl  $N_L$  der vorhandenen LEDs:

$$N_L = |\mathcal{L}|$$

Die Anzahl der LED Spalten wird als  $N_L^c$  und die der Zeilen als  $N_L^r$  bezeichnet. Generell ist der Ursprung der Nummerierung immer oben links und beginnt bei 0. Dabei entspricht  $c$  der horizontalen und  $r$  der vertikalen Dimension des Displays und beschreiben die LED-Spalten und Zeilen. Die lineare Indizierung inkrementiert von links nach rechts und von oben nach unten, zeilenweise und deckt sich damit mit dem Bilddatenfluss in der Pixelpipeline wie in Abschnitt 2.6 beschrieben. In der Abbildung 4.3 bezeichnet

$$\text{LED}_l, \forall l \in \{0, \dots, N_L - 1\}$$

die LED mit dem linearen Index  $l$  (blau dargestellt). In diesem Fall wird von der *LED Nummer* gesprochen. Alternativ kann eine LED durch ihre Zeilen und Spalten bestimmt werden:

$$\text{LED}_{r,c}, \forall (r \in \{0, \dots, N_L^r - 1\} \wedge c \in \{0, \dots, N_L^c - 1\})$$

An dieser Stelle sei angemerkt, dass Matlab das erste Element eines Objektes (z.B. einer Matrix oder Arrays) mit 1 indiziert und anstelle von 0. Für diesen Fall lässt sich mit

$$l = (r - 1) \cdot N_L^c + c \tag{4.1}$$

der lineare Index  $l$  für gegebene Koordinaten  $(r, c)$  berechnen.

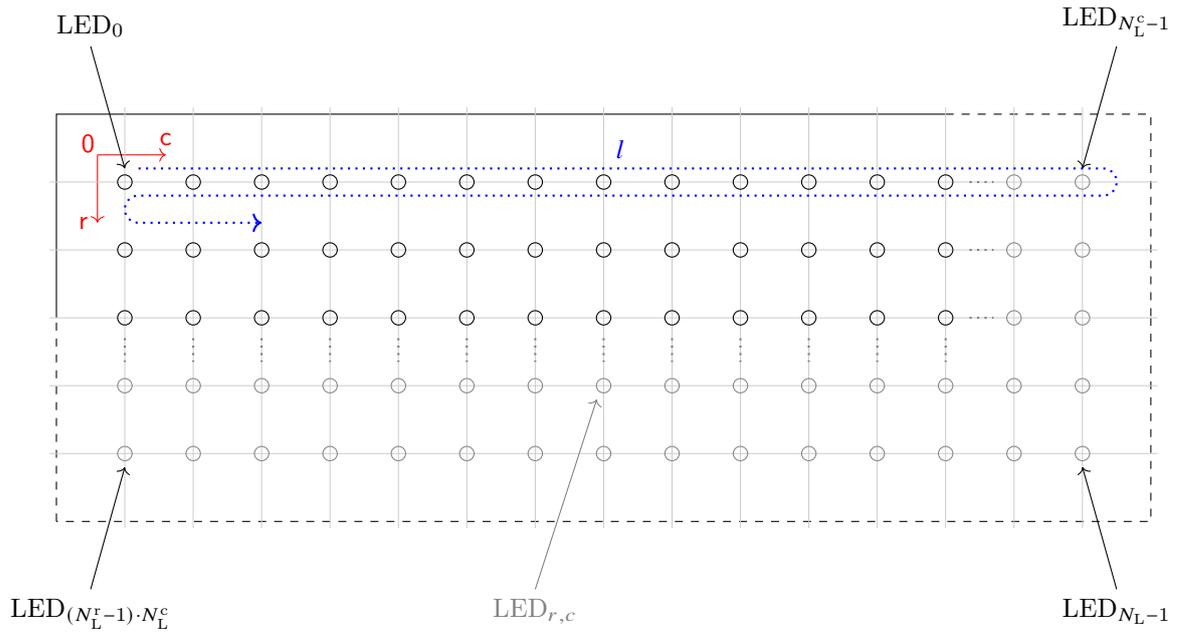


Abbildung 4.3.: LED Nummerierung und Anordnung.

#### 4.1.1. Vergleich mit niedrig aufgelösten BLUs

In Abschnitt 3.5.1 wurden bereits niedrig aufgelöste LSFs von Edge-Lit Geräten gezeigt. Zum Vergleich mit Direct-Lit BLUs wird in ein 6,1 Zoll Smartphone Panel der Firma *Huawei Technologies Co., Ltd.* verwendet. Dieses 16:9 Panel besitzt 7 LEDs an der unteren Kante und wurde im Rahmen der *Display Week 2015* im *German Pavilion* in San José (USA) mit einem Local-Dimming Algorithmus mit einer globalen Präprozessorstufe präsentiert [SSX15].

Die Direct-Lit BLUs mit 133 bzw. 319 LEDs wurde von *Continental* für den Einsatz im 12,3 Zoll ICD mit Seitenverhältnis 8:3 entworfen. In Abbildungen 4.4 bis 4.6 links sind die Leuchtdichteverteilungen der genannten BLUs dargestellt. Diese werden als sogenanntes *All-On* bezeichnet und werden bei maximaler LC-Transmission ( $t = 1$ ) und voll eingeschalteten LEDs gemessen und wird als

$$\mathbf{L}_{\Sigma,t}^{\text{abs}} \in \mathbb{R}_{0,+}^{N_P^r \times N_P^c} \text{ mit } t = 1$$

abgekürzt. Wenn nicht anders angegeben, wurden die Leuchtdichteverteilungen mit der Eldim UMaster [ELD] gemessen. Die LSF einer einzelnen LED  $l$  wird als

$$\mathbf{L}_l^{\text{abs}} \in \mathbb{R}_{0,+}^{N_P^r \times N_P^c}, \forall l \in \{0, \dots, N_L - 1\} \quad (4.2)$$

dargestellt (siehe Formel (2.14) auf Seite 32). Beispielhaft sind in den oben genannten Abbildungen zusätzlich zum All-On jeweils eine Leuchtdichteverteilung einer einzelnen LED des entsprechenden Panels abgebildet. In der Abbildung unterscheiden sich die maximalen Leuchtdichten zwischen den einzelnen Panels, was jedoch für die Untersuchung des Einflusses nicht relevant ist. Direkt ersichtlich ist jedoch, dass die LSF einer BLU einen sehr großen Einfluss auf die resultierende Homogenität hat. Darauf wird Abschnitt 4.2.1 näher eingegangen.

#### 4.1. Untersuchung des Einflusses auf die Pixel und Einfluss auf Local-Dimming

Für die Berechnung der LED-Werte wird statt der Leuchtdichte, der Einfluss, also der prozentuale Beitrag einer LED zu der Gesamtleuchtdichte in einem bestimmten Pixel benötigt (siehe Abschnitte 2.5.2 und 3.5.1 sowie Formeln (2.15) und (2.16) auf Seite 33). Der Einfluss wird beschrieben als:

$$\check{L}_l := \left( \check{L}_{(i,j),l} \right), \text{ mit } \check{L}_l \in [0; 1]^{N_P^r \times N_P^c}$$

und es gilt

$$\sum_{l=0}^{N_L-1} \check{L}_l = \mathbb{1}^{N_P^r \times N_P^c}$$

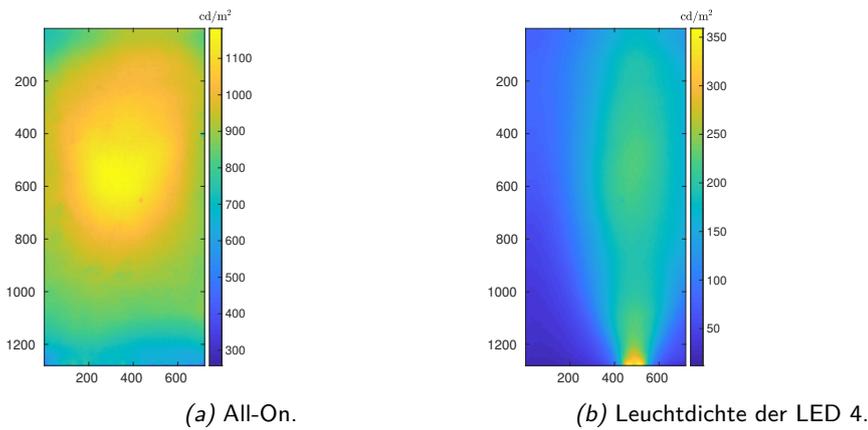


Abbildung 4.4.: Edge-Lit mit 7 LEDs an der Unterkante.

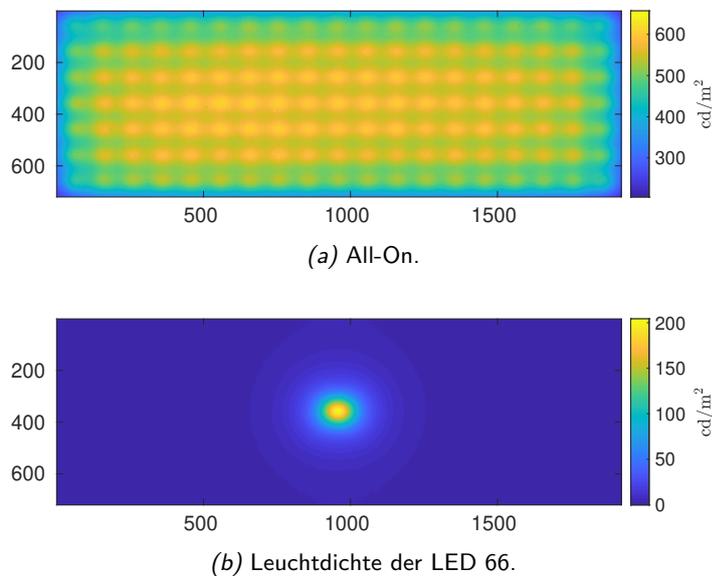


Abbildung 4.5.: Direct-Lit mit 133 LEDs (7 × 19).

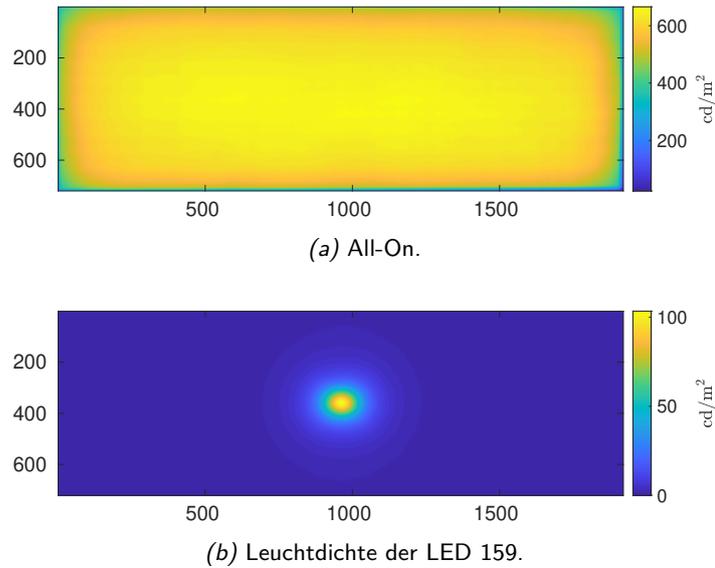


Abbildung 4.6.: Direct-Lit mit 319 LEDs (11 × 29).

Beispielhaft sind in Abbildung 4.7 die Einflüsse  $\check{L}_l$  der ausgewählten LEDs dargestellt. In den folgenden Abbildungen, wird das Edge-Lit Beispiel rotiert, wie in Abbildung 4.7a zu sehen. Aus Gründen der Übersichtlichkeit, sind die Maximalwerte der Skala unterschiedlich gewählt. Da sie sich zu sehr unterscheiden wäre der Einfluss der unteren Abbildungen mit diesem Falschfarbenschema nur schwer im Vergleich darzustellen. Zur Visualisierung dieses Einflusses sind in Abbildung 4.8 die vertikalen Querschnitte an den entsprechenden Markierungen aus Abbildung 4.7 zu sehen. Hierbei ist die Ordinatenachse logarithmisch dargestellt um die kleineren Einflüsse miteinander vergleichen zu können. Zu erkennen ist der maximale Einfluss der Edge-Lit LED von  $\approx 50\%$ , der der zentralen LED der 133er BLU  $\approx 32\%$  und  $\approx 15\%$  bei der 319er BLU. Aus der Abbildung geht ebenfalls hervor, dass der Einfluss der Direct-Lit LEDs in diesen Fällen drastisch mit der Entfernung von dem LED-Zentrum sinkt. Im Gegensatz dazu bleibt der Einfluss bei größerer Entfernung bei der Edge-Lit BLU hoch. Somit wird klar, dass eine LED in diesem Fall einen sehr globalen Beitrag zu der Gesamtleuchtdichte leistet. Je nach Bildinhalt muss dann zur Erhaltung der gewünschten Leuchtdichte in einem weiter entfernten Pixel, der Local-Dimming Algorithmus die LED-Werte globaler erhöhen. Dadurch werden jedoch auch weite Bildbereiche beleuchtet, welche die resultierende Leuchtdichte nicht benötigen. Der Verbesserung des Kontrastes kann in so einem Fall also deutlich geringer ausfallen.

### Maximaler Einfluss pro Pixel

Im Folgenden wird der maximale Einfluss pro Pixel untersucht. Dafür muss zunächst jedem Pixel  $P_p \in \mathcal{P}$  eine  $LED_l \in \mathcal{L}$  zugeordnet werden. Dabei ist der Einfluss in  $P_p$  maximal. Die Funktion  $\max_n^m(\cdot)$  soll von allen  $n$  Werten das Maximum liefern ( $m = 1$ ). Für  $m = 2$  liefert diese Funktion den zweithöchsten Wert usw.

$$\forall P_p \exists! LED_l : \left( \check{L}_{p,l} = \max_l^1 \check{L}_{p,l} \right) \text{ mit } l \in \{0, \dots, N_L - 1\} \wedge p \in \{0, \dots, N_P - 1\} \quad (4.3)$$

Die entsprechend nach Formel (4.3) gewählten LEDs, werden auch als *Haupt-LED* oder *Main-LED* bezüglich Pixel  $P_p$  bezeichnet und mit  $LED_{p,l}^M$  abgekürzt.

#### 4.1. Untersuchung des Einflusses auf die Pixel und Einfluss auf Local-Dimming

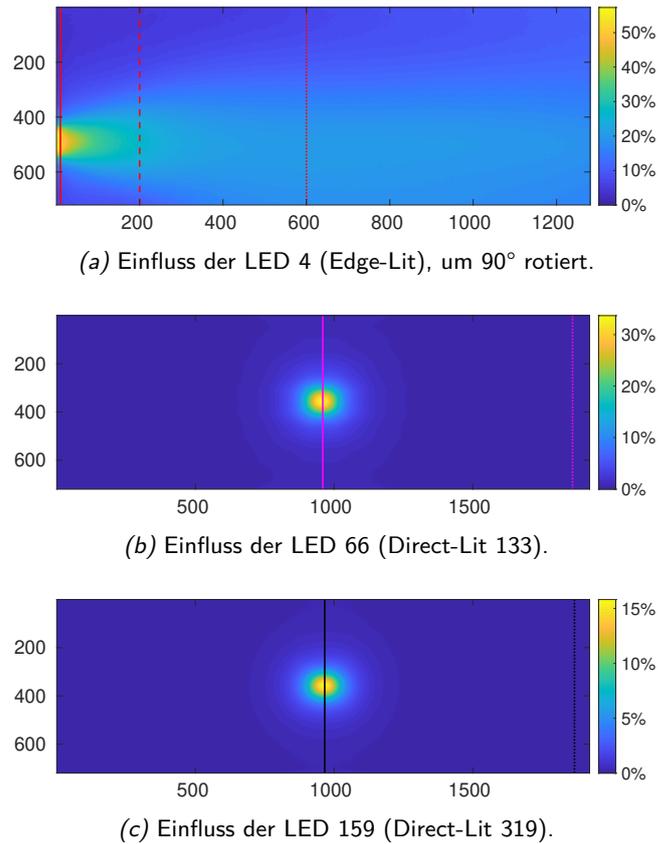


Abbildung 4.7.: Resultierende Einflüsse zu Abbildungen 4.4 bis 4.6.

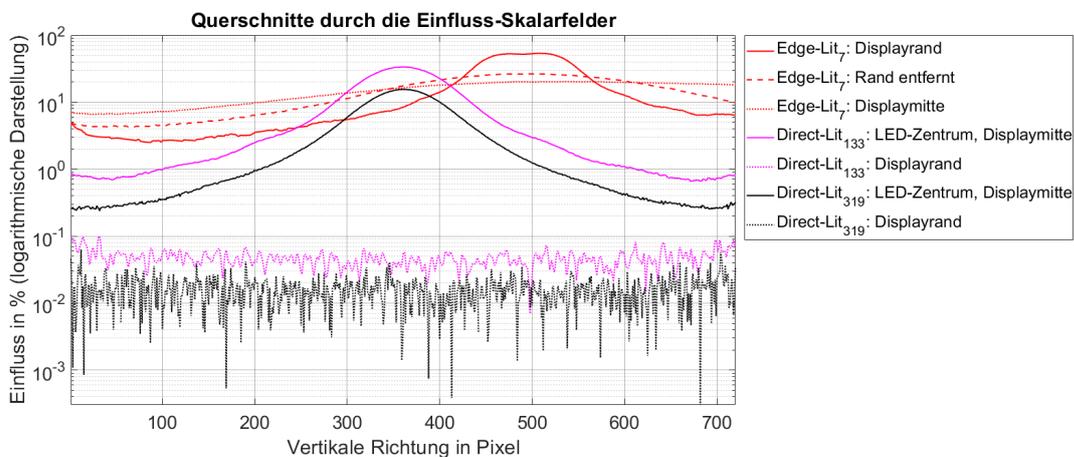


Abbildung 4.8.: Querschnitte durch die Einflüsse aus Abbildung 4.7.

Der maximale Einfluss der Haupt-LED ist wie später gezeigt wird, eine wichtige Local-Dimming Größe bzw. Kennzahl. Der Unterschied zwischen Edge-Lit und Direct-Lit BLUs wird bei der Zuordnung der Haupt-LED nochmal deutlich. In Abbildung 4.9 sind die Zuordnungen zu den einzelnen Pixeln abgebildet. Beispielhaft ist die Häufigkeit der Zuordnung einer Haupt-LED zu einem Pixel des 133er Panels in Abbildung 4.10 dargestellt. Dabei ist die Ordinatenachse mit der Anzahl  $|\mathcal{P}|$  normiert. Auffällig bei diesem Histogramm sind

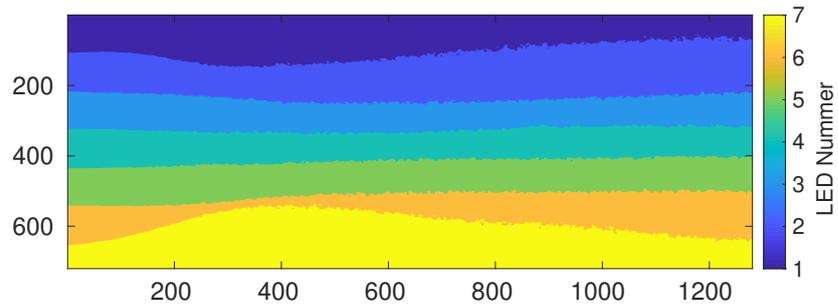
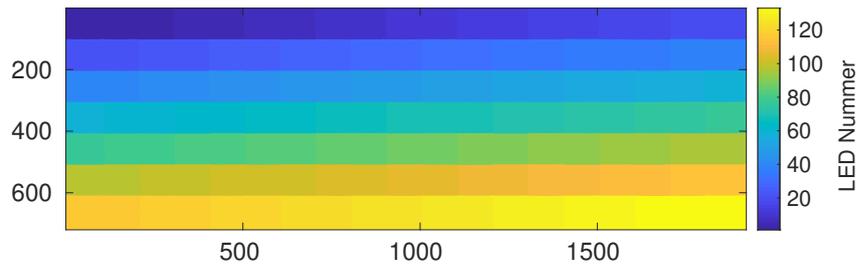
(a)  $LED_{p,l}^M$  beim Edge-Lit Gerät.(b)  $LED_{p,l}^M$  beim Direct-Lit Gerät.

Abbildung 4.9.: Visualisierung der Haupt-LED Zuordnung beim Edge-Lit Gerät mit 7 LEDs und Direct-Lit mit 133 LEDs.

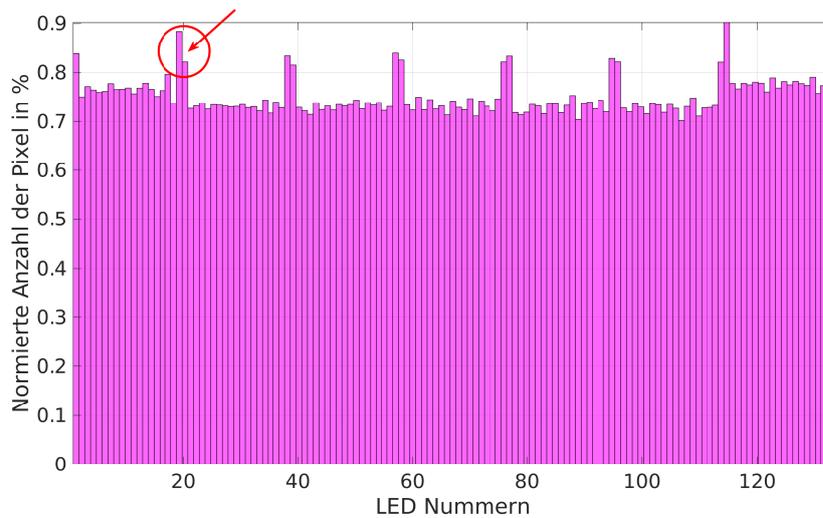


Abbildung 4.10.: Häufigkeit der Haupt-LED Zuordnung beim 133er Direct-Lit.

die wiederkehrenden Spitzen (rot markiert). Diese Häufigkeitsspitzen betreffen die LEDs an den Rändern der BLU, da aufgrund der Abwesenheit von LEDs außerhalb des Gerätes, der Einfluss der Haupt-LEDs an den Rändern deutlich höher liegt. Diese Spitzen treten also bei LED-Zeilenwechsel auf. Dies wird in Abbildung 4.11 deutlich. Im Vergleich zum Edge-Lit wird klar, dass die Einflüsse auf die Pixel aufgrund der symmetrisch positionierten LEDs, strukturiert sind. Dadurch sind die Local-Dimming Ergebnisse für ein um  $180^\circ$  gedrehtes Bild bei Direct-Lit BLUs symmetrisch, während sie es für ein Edge-Lit Gerät nicht

#### 4.1. Untersuchung des Einflusses auf die Pixel und Einfluss auf Local-Dimming

sind. Somit ist die Reduktion des Leistungsumsatzes beim Direct-Lit *invariant* gegenüber Rotation des Bildinhalts von  $n \cdot 180^\circ, \forall n \in \mathbb{Z}$ . Die Verteilung des maximalen Einflusses basiert auf Abbildung 4.11 wird in Abbildung 4.12 zum Vergleich als Histogramm dargestellt. Dabei beträgt die Auflösung der Bins 0,5%. Es wird deutlich, dass die Standardabweichung  $\sigma$  für höher werdende LED Anzahl geringer wird. Insbesondere beim Edge-Lit Geräten werden sehr hohe Spitzen Einflüsse ( $> 70\%$ ) erreicht.

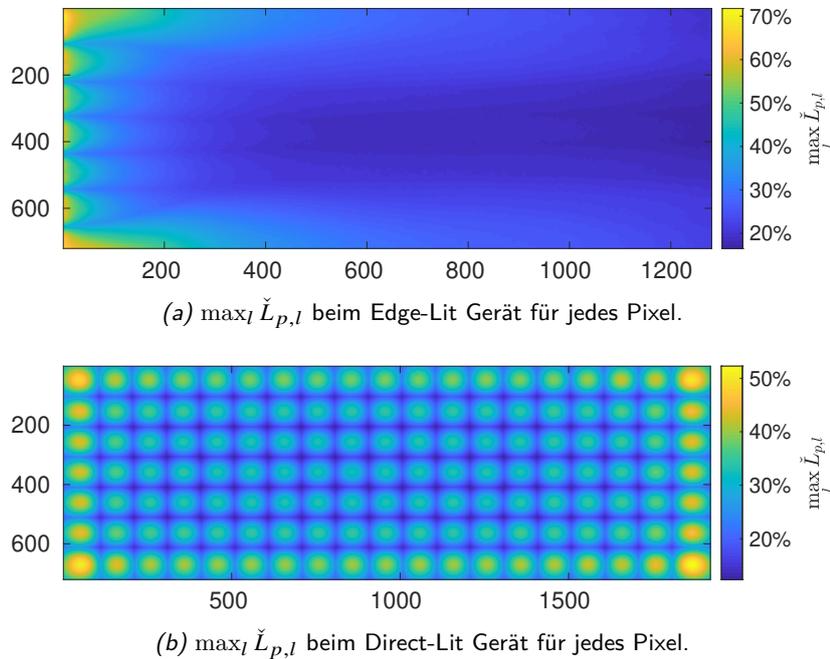


Abbildung 4.11.: Maximale Einflüsse der Haupt-LEDs pro Pixel beim Edge-Lit Gerät mit 7 LEDs und Direct-Lit mit 133 LEDs.

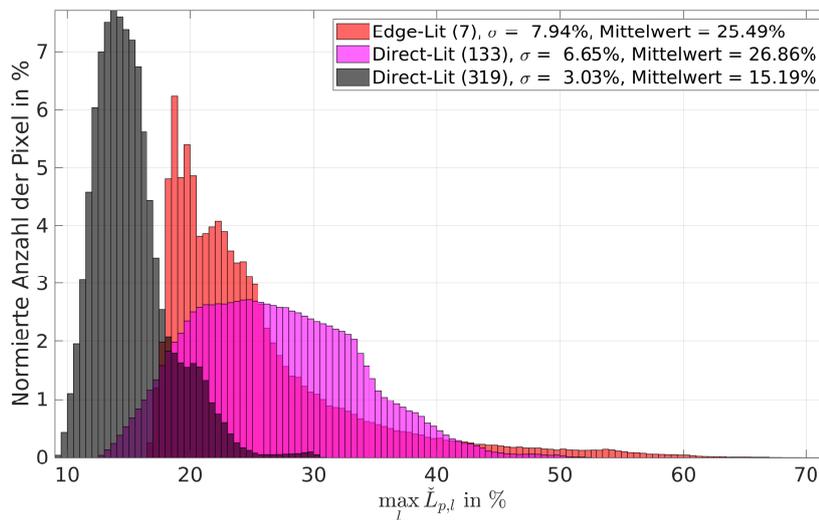


Abbildung 4.12.: Verteilung der maximalen Einflüsse mit einer Bin-Weite von 0,5%.

## Fazit

Es lässt sich schlussfolgern, dass bei hochauflösenden BLUs nicht jede einzelne LSF der LEDs gespeichert werden muss wie es bei Edge-Lit der Fall ist, da es viele sehr ähnliche LSFs in gleicher oder gespiegelter Ausrichtung gibt. Dazu wird in Abschnitt 5.3 eine neue Methode vorgestellt, welche in dieser Arbeit entwickelt wurde.

## Lokalität

Bereits in Abbildung 4.2 wurde der Begriff der Lokalität in Zusammenhang mit Local-Dimming erwähnt. Der Begriff wird hier definiert, um ihn zu quantifizieren. In diesem Abschnitt wird dazu eine Berechnungsvorschrift vorgestellt. Es soll gezeigt werden, dass insbesondere für Edge-Lit BLUs die Lokalität stark abnimmt, jedoch für höher werdende LED-Zahlen durch eine LSF Eigenschaft bzw. Größe dominiert wird.

Dazu repräsentiere die Matrix bzw. der Vektor  $\check{L}^{\max} \in [0; 1]^{1 \times N_P}$  den maximalen Einfluss in jedem linear indizierten Pixel  $P_p$  an der Stelle  $p$ :

$$\check{L}^{\max} := \left( \max_l \check{L}_{p,l} \right) = \left( \check{L}_p^{\max} \right)$$

Der höchste maximale Einfluss sowie der mittlere Einfluss von  $\check{L}^{\max}$  werden mit

$$\check{L}_{\text{total}}^{\max} = \max(\check{L}^{\max}) \quad (4.4)$$

$$\bar{\check{L}}^{\max} = \frac{1}{N_P} \cdot \sum_{p=0}^{N_P-1} \check{L}_p^{\max} \quad (4.5)$$

dargestellt. Der gesamte mittlere Einfluss  $\bar{\check{L}}$  von allen LEDs pro Pixel leitet sich wie folgt her:

$$\begin{aligned} \bar{\check{L}} &= \frac{1}{N_L \cdot N_P} \cdot \sum_{p=0}^{N_P-1} \underbrace{\sum_{l=0}^{N_L-1} \check{L}_{p,l}}_{=1, \forall p \in \{0, \dots, N_P-1\}} \\ &\stackrel{\text{mit Formeln (2.15), (2.16)}}{=} \frac{1}{N_L \cdot N_P} \cdot \sum_{p=0}^{N_P-1} 1 \\ &= \frac{1}{N_L \cdot N_P} \cdot N_P \\ &= \frac{1}{N_L} \end{aligned} \quad (4.6)$$

und entspricht dem Kehrwert der LED-Anzahl. Mithilfe von Formeln (4.4) bis (4.6) kann dann der Begriff *Lokalität* einer BLU als Funktion definiert  $\check{\Lambda}$  werden  $\forall N_L \in \mathbb{N} : N_L > 1$ :

$$\check{\Lambda} \left( \Delta_{\check{L}^{\max}}^{\check{L}_{\text{total}}^{\max}}, N_L \right) = 1 - \left( \check{L}_{\text{total}}^{\max} - \bar{\check{L}}^{\max} \right) \cdot \left( 2 - e^{-\bar{\check{L}}} \right) = 1 - \underbrace{\left( \check{L}_{\text{total}}^{\max} - \bar{\check{L}}^{\max} \right)}_{\Delta_{\bar{\check{L}}^{\max}}^{\check{L}_{\text{total}}^{\max}}} \cdot \left( 2 - e^{-1/N_L} \right) \quad (4.7)$$

In Formel (4.7) geht die LED-Anzahl exponentiell ein und dient als Gewichtung von  $\Delta_{\check{L}^{\max}}^{\check{L}_{\text{total}}^{\max}}$ . Die höchste Lokalität von 1 wird bei sogenanntem *Pixeldimming* erreicht. Dabei wird beispielsweise mit einem zweiten

LC-Panel die Leuchtdichte hinter jedem Pixel direkt moduliert [Che+17d]. Im Optimalfall ist  $\check{L}_{total}^{max}$  maximal und entspricht  $\check{L}$ , womit  $\check{\Lambda} = 1$  ist. Somit ergibt sich für OLED-Displays ebenfalls die höchstmögliche Lokalität. Allgemein gilt für eine höhere LED-Anzahl:

$$\lim_{N_L \rightarrow N_P} \check{\Lambda} \left( \Delta_{\check{L}_{max}}^{\check{L}_{total}^{max}}, N_L \right) = \lim_{N_L \rightarrow N_P} 1 - \Delta_{\check{L}_{max}}^{\check{L}_{total}^{max}} \cdot \underbrace{\left( 2 - e^{-1/N_L} \right)}_{\rightarrow 1} = 1 - \Delta_{\check{L}_{max}}^{\check{L}_{total}^{max}}$$

wobei für eine hohe Lokalität  $\Delta_{\check{L}_{max}}^{\check{L}_{total}^{max}}$  minimiert werden soll. Für Global-Dimming sei die Lokalität  $\check{\Lambda} = 0$ . Anhand Abbildung 4.11b wird deutlich, dass  $\check{\Lambda}$  ansteigt, je homogener die Visualisierungen von  $\check{L}^{max}$  werden. In Abbildung 4.13 ist eine Funktionenschar der Funktion  $\check{\Lambda}$  für  $N_L \in \{2, \dots, 500\}$  sowie die Lokalitäten der bis jetzt betrachteten BLUs. Dabei steht das  $\Delta$  in der Abbildung für  $\Delta_{\check{L}_{max}}^{\check{L}_{total}^{max}}$ . Es ist zu erkennen, dass das der exponentielle Faktor der Funktion die Lokalität im Bereich der wenigen LED-Anzahl sehr stark beeinflusst. In diesen Bereich fallen die Edge-Lit BLUs. Bei weiterer Steigerung der LED Zahl (Direct-Lit) spielt  $\Delta_{\check{L}_{max}}^{\check{L}_{total}^{max}}$  die entscheidende Rolle. Für Pixeldimming und OLED-Displays beträgt die Lokalität wie weiter oben beschrieben 100 % (rote Linie). Die genauen  $\check{\Lambda}$  Werte der Prototypen sind in Tabelle 4.1 aufgeführt.

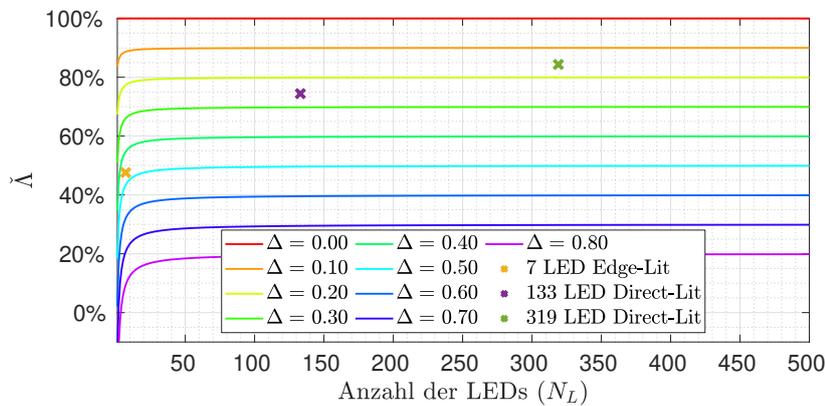


Abbildung 4.13.: Lokalitäten mit steigender LED Anzahl und verschiedene  $\Delta_{\check{L}_{max}}^{\check{L}_{total}^{max}}$ .

	BLUs ( $N_L$ )		
	7	133	319
$\check{\Lambda}$	47,58 %	74,42 %	84,35 %

Tabelle 4.1.: Vergleich der resultierenden  $\check{\Lambda}$  angegeben in %.

Für den Local-Dimming Algorithmus lässt sich schlussfolgern, dass die Ergebnisse bezüglich der Reduktion des Leistungsumsatzes (*Powersaving*) mit steigender Lokalität besser werden. Wie in Abschnitt 3.1 gezeigt wurde, ergibt sich dieses erhöhte Powersaving unter anderem durch die im automobil Bereich typischen Bildinhalte, insbesondere bei ICDs. Außerdem wird das *Halo*, welches in Abschnitt 6.2 behandelt wird, verringert und führt zur besseren Bildqualität und somit zur erhöhter Wertigkeit des Displays. Es wird auch

deutlich, dass Edge-Lit deutliche Nachteile beim Dimmen von speziellen HMI Bildern gegenüber Direct-Lit Displays aufweist. Deswegen wird in dieser Arbeit an Local-Dimming für Direct-Lit BLUs geforscht.

#### 4.1.2. Verschiedene LSF-Formen

In den vorangegangenen Abschnitten wurden bereits zwei verschiedene Direct-Lit BLUs und eine Edge-Lit BLU vorgestellt. Ab hier wird, wenn nicht anders angegeben, immer von Direct-Lit BLUs gesprochen. Im Folgenden werden weitere LSFs vorgestellt.

Die Abbildung 4.14 zeigt das All-On eines 16:9 TV-Gerätes (Hisense 65XT910) mit 240 Dimming Zonen sowie einen Querschnitt in horizontaler Richtung. Dabei ist die Leuchtdichteskala der Falschfarbendarstellung in  $\text{cd m}^{-2}$  angegeben. In Abbildung 4.14b sind neben dem Querschnitt durch das All-On auch die LED-Positionen

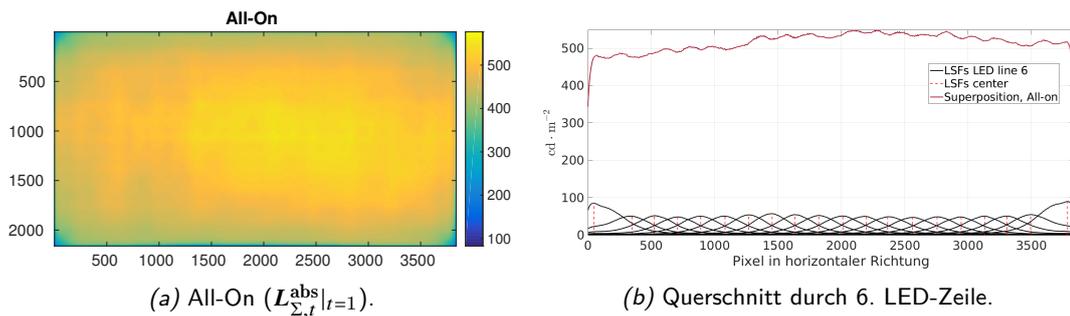


Abbildung 4.14.: Backlight des Hisense 65XT910 mit  $12 \times 20$  LEDs.

abgebildet. Der sehr geringe Einfluss einer einzelnen LED ist ersichtlich. Dabei beträgt  $\bar{L}^{\max} = 11\%$ . Ebenso erkennbar ist, dass der Abstand der Rand LEDs im Vergleich zu den übrigen nicht äquidistant ist. Zusätzlich ist deren Leuchtdichte fast doppelt so hoch. Hierbei war das Ziel der BLU Designer, die Lichtverteilung – auch am Rand – möglichst homogen zu gestalten und wurde mithilfe höheren Strömen realisiert. Trotz des hochpreisigen Gerätes, sind die Leuchtdichteabfälle insbesondere an den Ecken jedoch mit bloßem Auge zu erkennen und sind auch in der Falschfarbendarstellung sichtbar. In Abbildung 4.15 sind 3 der 240 Dimming Zonen mit demselben PWM-Tastgrad von 100% dargestellt, wobei sich die Leuchtdichte Unterschiede sehr gut erkennen lassen.

Bekanntlich können Stromvariationen in den einzelnen LEDs einer BLU zu einem Farbdrift durch die Verschiebung des Weißpunktes führen [OSR; Dyb+05; BM10]. Folglich wird nicht empfohlen die LEDs mit Amplitudenmodulation (AM) zu dimmen.

Für diese BLU ergibt sich eine Lokalität anhand Formel (4.7) von 75% und liegt dabei sehr nahe bei der gezeigten 133er BLU welche deutlich weniger Dimming Zonen besitzt.

Bisher wurden LSFs betrachtet, welche in erster Näherung radialsymmetrische Lichtverteilungen aufweisen. Darüber hinaus gibt es einen Ansatz, die Kavitäten der BLU in denen die LEDs platziert sind so zu designen, dass die Leuchtdichteverteilung in einer rechteckige Form resultiert. In Abbildung 4.16 ist solch eine Verteilung abgebildet. Die roten Linien markieren die vertikalen Grenzen der rechteckigen Dimming Zonen. In dem dazugehörigen Querschnitt rechts ist die stufenförmige Verteilung ersichtlich und zeigt die resultierenden steilen Kanten an den Grenzen der Kavitäten. Diese gezeigte Verteilung basiert auf einer Simulation des Strahlenganges ausgehend von der raumwinkelabhängigen Lichtverteilung der LED durch den optischen Stack der BLU (siehe Abschnitt 2.3.1). Diese Simulationemethode wird als *Raytracing* bezeichnet [Sch06]. In diesem Fall besteht die Simulation aus  $400 \times 400$  Stützstellen und inkludiert 25 Dimming Zonen. Die

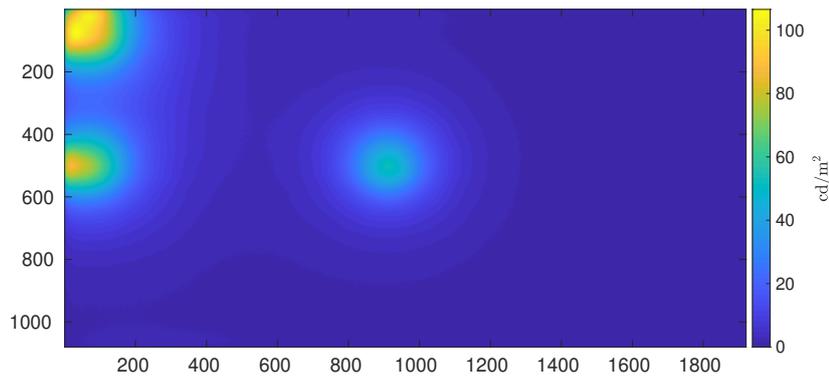


Abbildung 4.15.: Hisense 240er Backlight mit 3 aktivierten Dimming-Zonen.

resultierenden Stützstellen der Leuchtdichteverteilung wurden von einem Projektpartner zur Verfügung gestellt und wurde für eine BLU mit 216 Dimming Zonen für den Einsatz in einem ICD konzipiert.

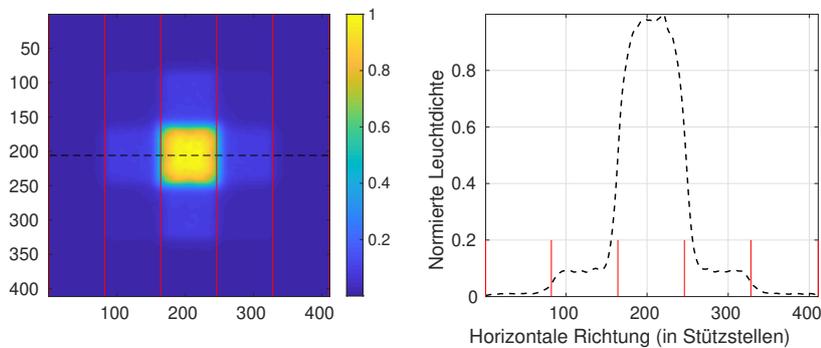


Abbildung 4.16.: Simulierte LSF für eine 8:3 BLU mit 216 LEDs. Dargestellt sind die Simulationsdaten einer einzelnen LED.

Mithilfe der Dimensionierung des LC-Panels (8:3 Seitenverhältnis mit 12,3 Zoll Diagonale) und des bekannten LED-Abstandes von 1,25 cm lässt sich die LSF der zentralen LED für dieses Panel generieren und wird in Abbildung 4.17 dargestellt. Die LED Positionen bzw. die Zentren der Kavitäten sind mit roten Punkten markiert. Diese BLU mit 240 LEDs wurde von einem Projektpartner zur Verfügung gestellt. Die auf den Leuchtdichtemessungen dieses Projektpartners basierenden Daten sind in Abbildungen 4.18 und A.3 dargestellt. Dabei sind in der Abbildung im Anhang 9 Einzelmessungen sowie die resultierenden Einflüsse visualisiert. Zur Erhöhung der Effizienz der LED Treiber, gibt es auch die Möglichkeit mehrere Dimming Zonen bzw. LEDs zu einer Einheit zusammenzufassen. Dieses Vorgehen wird als *Clustering* bezeichnet und in Abschnitt 5.5 erläutert. Als Beispiel ist in Abbildung 4.18 neben der Einzelmessung der zentralen LED auch eine Messung einer *virtuellen* zentralen LED mit simulierten 60 Dimming Zonen, sodass immer 4 physikalische Zonen zu einer virtuellen Zone gebündelt werden. Die entsprechenden Querschnitte entlang der markierten Linien sind unten im Bild dargestellt. Bei genauer Betrachtung der Leuchtdichtemessungen fallen die Moiré-Effekt typischen hochfrequente Intensitätsschwankungen negativ auf und reduzieren die Qualität der Messungen. Diese kommen aufgrund der ähnlichen Gitterstruktur bzw. Pixelstruktur des Sensors des Leuchtdichtemessers und der Pixelauflösung der LC-Panels zustande. Zur Vermeidung dieses Effektes wird

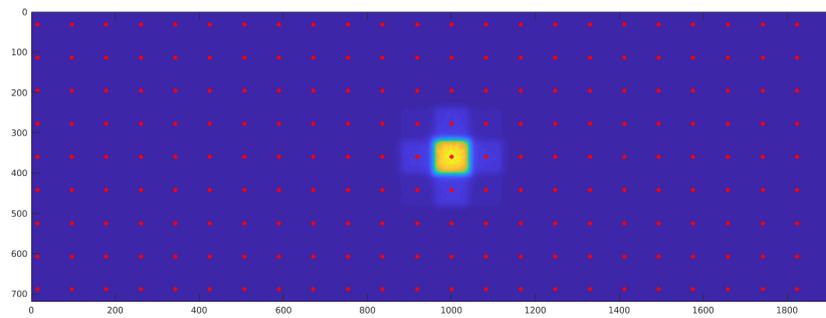


Abbildung 4.17.: Zentrale LED einer simulierten 216er BLU. Die LED-Positionen sind mit roten Punkten markiert.

empfohlen, den Fokus der Kamera etwas von der Oberfläche des Displays zu verschieben und somit leicht unscharf zu stellen [Bla16b]. Mit dieser Methode wird die Qualität der Messung entsprechend verbessert. In Abbildung 4.19 ist eine Fotografie dieser BLU gezeigt. Das LC-Panel wurde hier weggelassen. Dargestellt ist ein Schachbrettmuster mit alternierend 100 % bzw. 0 % PWM-Tastgrad. Eine Ähnlichkeit zu Abbildung 4.17 ist auf dem Foto zu erkennen. Auch in den zur Verfügung gestellten Messungen ist bei genauer Betrachtung eine rechteckige Form der Dimming Zonen noch erkennbar.

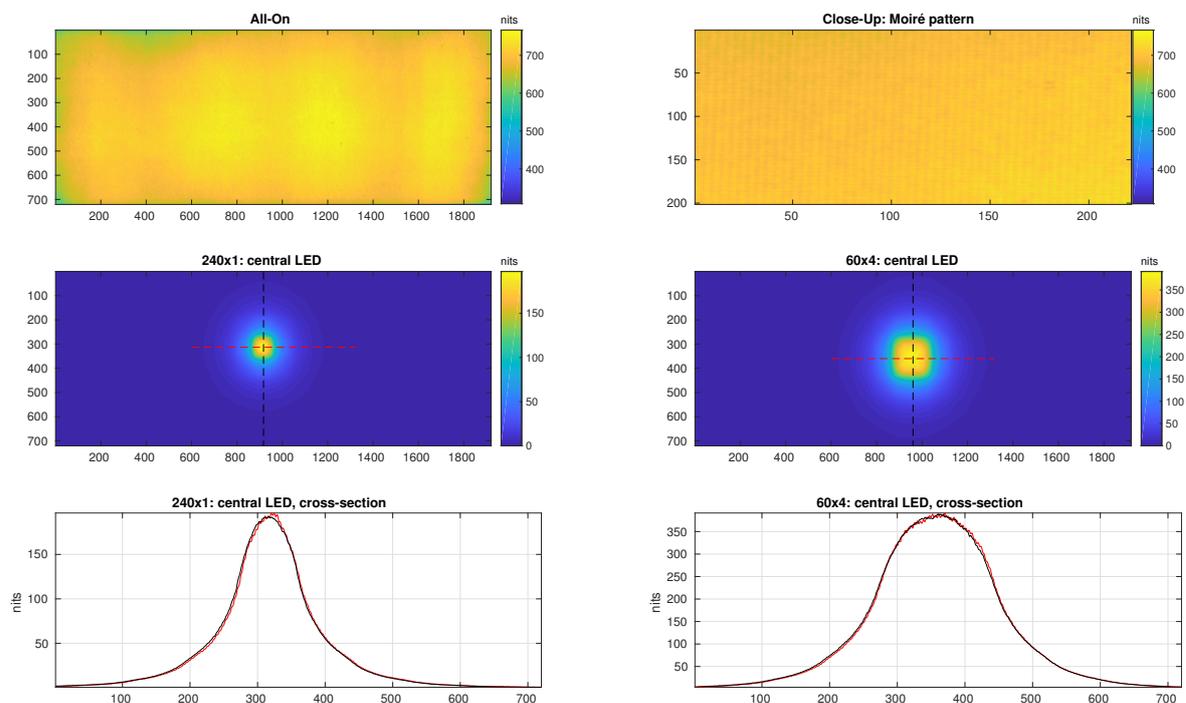


Abbildung 4.18.: Messungen des All-Ons sowie der zentralen Dimming Zone, gemessen vom Projektpartner. Dargestellt sind auch die markierten Querschnitte.

Die resultierenden Einflüsse und Querschnitte sind in Abbildung 4.20 nochmal für die dritte LED Zeile dargestellt. Die aus den Messungen abgeleiteten maximalen Einflüsse pro Pixel ( $\check{L}_p^{\max}$ ) für diese BLU liefert nach Formel (4.7) eine Lokalität  $\check{\Lambda}$  von 81,37 % und ordnet sich damit in den Bereich der 319er BLU von *Continental* ein (siehe Tabelle 4.1). Aus diesem Grund wird erwartet, dass diese BLU etwa ähnlich gute

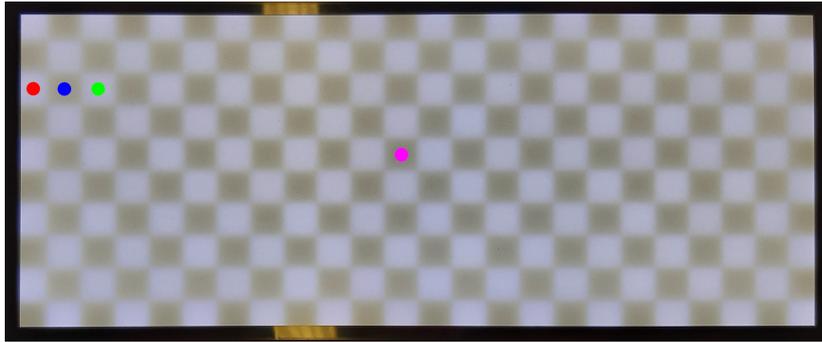


Abbildung 4.19.: Foto der obigen BLU (ohne LC-Panel) mit Schachbrettmuster (10 × 24 Zonen) und Markierung für Abbildung 4.20.

Ergebnisse liefern wird, wie die 319er BLU. Im Gegensatz dazu beträgt die Lokalität  $\check{L}$  für diese BLU mit 60 virtuellen Dimming Zonen 65,63 %, sodass ein schwächeres Local-Dimming Ergebnis zu erwarten ist. In diesem Abschnitt wurden bereits LSFs mit 240 Dimming Zonen vorgestellt. Dabei handelte es sich um ein

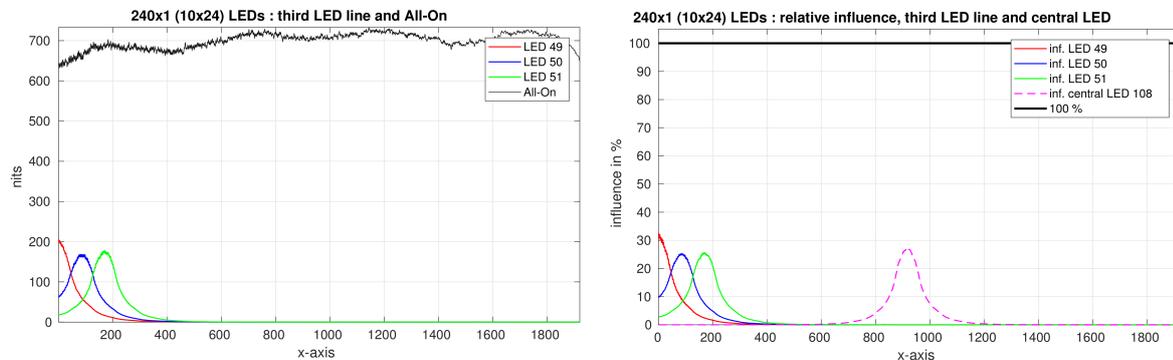


Abbildung 4.20.: Übersicht und Querschnitte durch LSF und resultierende Einflüsse der obigen 240er BLU.

BLUs eines TV-Gerätes von *Hisense* sowie eines ICD von *Continental*. Bedingt durch das unterschiedliche Seitenverhältnis der beiden BLUs ist eine Anordnung von 12 × 20 bzw. 10 × 24 Dimming Zonen gegeben. Unabhängig von dem Seitenverhältnis ist die Charakteristik der beiden BLUs verschieden. Zur besseren Veranschaulichung sind in Abbildung 4.21 jeweils die Querschnitte durch die Einflüsse einer zentral liegenden LSF logarithmisch dargestellt. Zusätzlich zum Verdeutlichen des unterschiedlichen LED-Abstandes jeweils mit vertikalen, farblich gestrichelten Linien, sind die Position der 8 nächsten LED-Zentren in horizontaler Richtung angedeutet. Die schwarz gestrichelte, horizontale Linie an der Stelle 960 ist dabei das gemeinsame Zentrum der im Vergleich stehenden LSFs.

Es ist direkt ersichtlich, dass der maximale Einfluss sich gravierend zwischen den beiden LSFs unterscheidet. Bei dem TV liegt er bei  $\approx 10\%$ , während er bei dem ICD bei  $\approx 25\%$  liegt. Auch erkennbar ist der schnellere bzw. steilere Abstieg des Einflusses der blauen Kurve. So erreichen beide im Bereich der zweiten Nachbarzone schon einen Einfluss von unter 2%, wobei die blaue Kurve deutlich schneller gegen 0 fällt. Der rote Graph zeigt, wie *global* sich die TV BLU verhält. In Kombination mit  $\check{L}_{total}^{max}$  folgt somit, dass die blaue Kurve resultierend aus der LSF von *Continental* deutlich effizienter bzgl. Local-Dimming sein wird, als die *Hisense* TV BLU. Durch den schnelleren Abfall können die LEDs *lokaler* an den Bildinhalt angepasst werden.

Besteht das Bild aus vielen größeren schwarzen Flächen und nur wenigen hellen Spots, wird auch der Schwarzwert verbessert, da nur wenige Dimming Zonen um den Spot herum eingeschaltet werden müssen um die erforderliche Leuchtdichte zu erzeugen. Jedoch kann solch eine LSF das *Halo* Artefakt negativ beeinflussen. Darauf wird in Abschnitt 6.2 eingegangen.

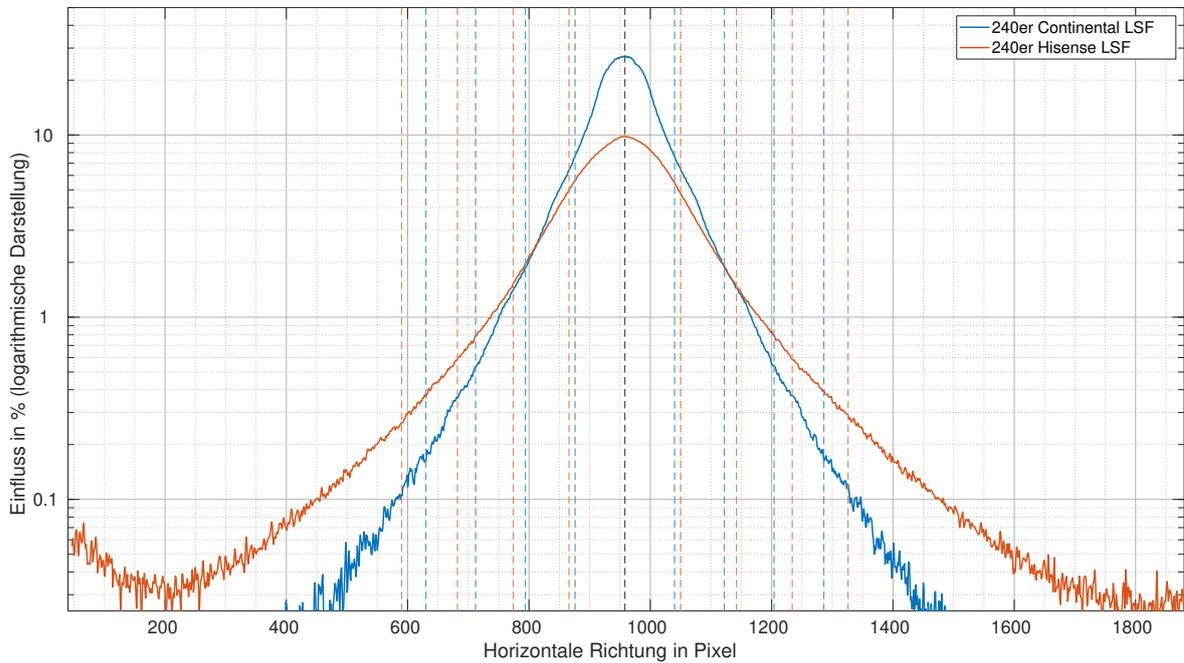


Abbildung 4.21.: Vergleich des Einflusses der beiden vorgestellten 240er BLUs mit Markierung der LED Zentren in horizontaler Richtung.

Bereits in [LCT07; Che+08] wird postuliert, dass quadratische LSF Formen eine gute bzw. die beste Lösung darstellen. Dabei soll der Einfluss in benachbarte Dimming Zonen 0 betragen, d.h. dass das Übersprechen (*Crosstalk*) komplett unterdrückt wird. Da solch eine Realisierung nicht möglich ist, wurde in den genannten Veröffentlichungen eine Gauß-Verteilung höherer Ordnung (engl. super-Gaussian function) als Grundlage für die Simulation angenommen um sich der optimalen quadratischen Verteilung zu nähern und realistische Ergebnisse zu erhalten. Für den eindimensionalen Fall lässt sich diese Verteilung als Funktion

$$f(x) = A \cdot e^{-\left(\frac{(x-x_0)^2}{2\sigma^2}\right)^p} \quad (4.8)$$

beschreiben. Dabei ist  $x_0$  der Erwartungswert und  $\sigma^2$  die Varianz. Für ein  $p > 1$  ist  $f(x)$  eine Gauß-Verteilung höherer Ordnung.

Mit solchen rechteckigen Dimming Zonen wird eine höhere Homogenität insbesondere an den Rändern erwartet, da hier die Leuchtdichte optimalerweise nicht zu den Rändern hin abfällt. Dies wird in Abbildung 4.22 deutlich. Es werden die All-Ons sowie die Querschnitte der 133er, 319er und der zuletzt vorgestellten BLU mit 240 Dimming Zonen dargestellt. Die letzte BLU hat einen ähnlichen rechteckigen Charakter wie oben beschrieben, jedoch mit Crosstalk auf die benachbarten Zonen. Die Leuchtdichte fällt vergleichsweise geringer an den Rändern ab.

Diese vorgeschlagene LSF Form bringt jedoch auch Nachteile mit sich. In [LCT07; Che+08] wird dieses

#### 4.1. Untersuchung des Einflusses auf die Pixel und Einfluss auf Local-Dimming

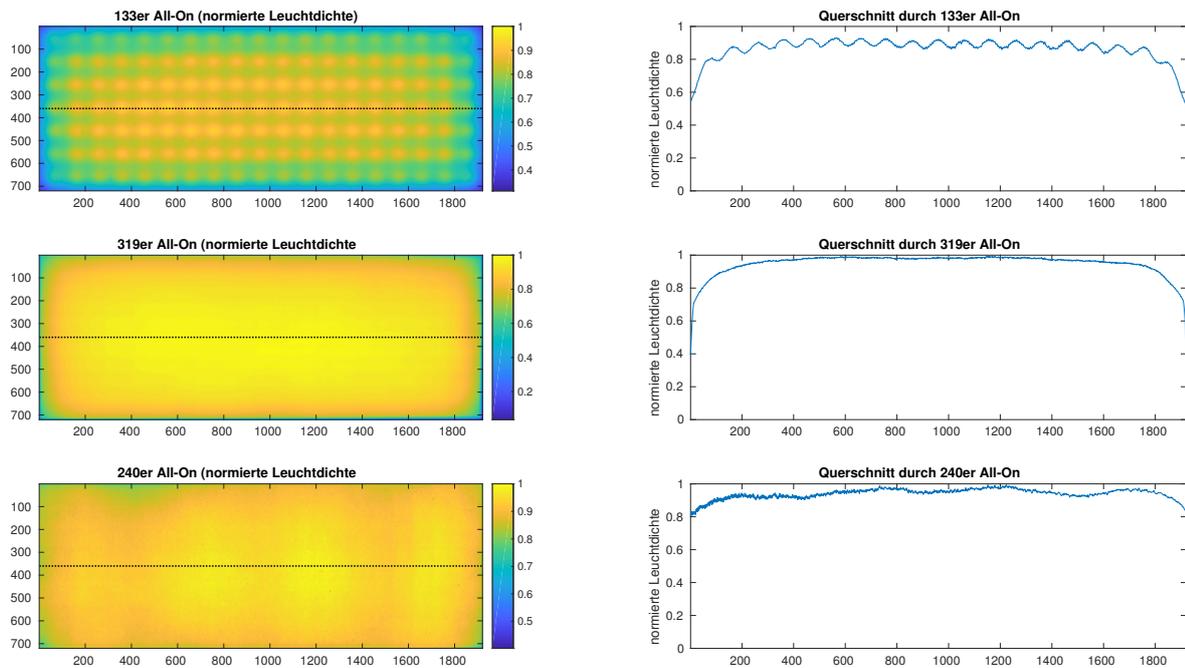
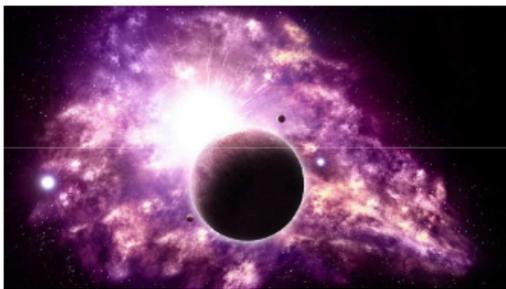
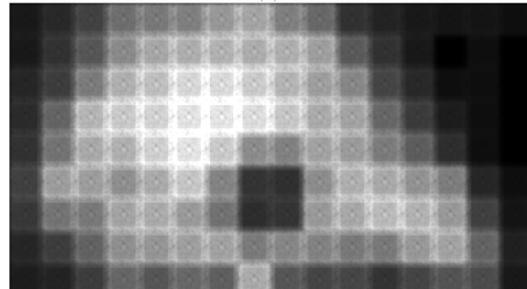


Abbildung 4.22.: Vergleich der vorgestellten 133er, 319er und 240 BLUs.

Problem deutlich. Dazu ist in Abbildung 4.23 ein Testbild gezeigt, sowie die resultierende Leuchtdichteverteilung berechnet mit einem Local-Dimming Algorithmus. Diese Simulation zeigt deutliche Gitter Muster. Diese Muster entstehen durch die Überlagerung des Lichtes einer Dimming Zone mit dem eingekoppelten Licht der Nachbarzonen. Da in diesen Arbeiten eine Gauß-Verteilung höherer Ordnung genutzt wurde und diese keine unendlichen Steigungen vorweisen kann, weicht das Modell von dem *optimalen* quadratischen Design ab und führt aufgrund der Auslegung des LED-Pitches (-Abstandes) basiert auf der quadratischen Verteilung zu dieser Superposition des Lichtes an den Grenzen der Zonen. In Abschnitt 4.5 werden diese Effekte anhand von radialen LSFs aufgezeigt.



(a) Testbild.



(b) Resultierende Hintergrundbeleuchtung nach Anwendung eines Local-Dimming Algorithmus.

Abbildung 4.23.: Testbild und resultierende Leuchtdichte basierend auf einer quadratischen LSF. Bilder entnommen aus [LCT07].

Zusätzlich zu den gezeigten Problemen der BLU selbst, kann ein weiteres Artefakt auftreten, welches durch die steilen Flanken der isolierten, quadratischen LSF begünstigt wird, nämlich die sichtbare *Blockbildung* bei

Anwendung eines Local-Dimming Algorithmus. Dieses Artefakt wird durch verschiedene Ursachen ausgelöst, welche zu einer inkorrekten Kompensation und schließlich zur Wahrnehmung der Dimming Zonen Grenzen führen können. Dazu zeigt Abbildung 4.24 beispielhaft die Entstehung solcher Blöcke nach der Anwendung von Local-Dimming. Die linke Seite der Abbildung zeigt den ungedimnten Fall, während die rechte die gedimmte Situation darstellt. Es sind 2 benachbarte, scharf getrennte Dimming Zonen ohne Crosstalk abgebildet. In diesem Fall sei der Inhalt des Bildes in den beiden Zonen zum größten Teil dunkelblau mit jeweils ein paar maximal rote bzw. sehr dunkle grüne Pixel. Zwecks Visualisierung sind die Abbildungen aufgehellt und stellen das resultierende Bild (Backlight, sowie die Modulation durch das LC-Panel) dar. Im unteren Teil der Abbildung sind die maximalen Post-Gamma Werte pro Dimming Zone als Balken dargestellt. Der Einfachheit halber wird zur Visualisierung ein Gammawert von  $\gamma = 1$  angenommen. Dabei sollen die Werte in diesem Beispiel eine 8 Bit Auflösung vorweisen. Die orange gestrichelte Linie zeigt den Verlauf des entsprechenden Backlights bzw. der Leuchtdichte (normiert). Mit Anwendung des Local-Dimming Algorithmus aus Abschnitt 3.5.2 ergibt sich die rechts dargestellte Situation.

Die Zone 1 kann aufgrund der roten Pixel nicht gedimmt werden und verbleibt somit im Originalzustand mit vollem Backlight und Kompensation. In der zweiten Dimming Zone hingegen, liegt der Maximalwert bei 2 von 255. Dadurch kann die entsprechende LED gedimmt und die Pixelwerte kompensiert werden. In diesem Fall ergäbe der Wert für blau 127,5 und läge somit zwischen zwei Werten bei einer Auflösung von 8 Bit. Dadurch wird der Wert auf- oder abgerundet und die TFTs entsprechend angesteuert. Somit ergibt sich für den Betrachter ein minimal helleres bzw. dunkleres Bild in dieser Zone als im ersten Dimming Bereich. Dieser Unterschied fällt durch die harte Grenze zwischen den Zonen dem Betrachter ins Auge und bildet dieses Artefakt. Generell ist bei solchen scharf abgetrennten Kavitäten die Gefahr von dem beschriebenen Artefakt höher als bei LSFs mit einem sanfteren Übergang in die Nachbarzonen.

Folgende Gegebenheiten können zur Blockbildung durch Abweichung von der Sollleuchtdichte und Farbdriftes führen:

- Geringe Auflösung der Post-Gamma Werte während der Prozessierung und TFTs führen zu Ungenauigkeiten bei der Kompensation. Ebenso eine zu geringe PWM Auflösung.
- Ungenauigkeiten der Gamma-Messung oder der De-Gamma Berechnung. Z.B. zu geringe Anzahl an Messungen oder Annahme einer einzigen Gammakurve für alle 3 Primärfarben.

## Schlussfolgerungen

In den obigen Abschnitten wurden verschiedene LSFs Formen vorgestellt und bzgl. ihrer Lokalität und Einfluss miteinander verglichen. Aus diesen Betrachtungen kann geschlussfolgert werden, dass neben dem Local-Dimming Algorithmus an sich, auch die BLUs mit ihren charakteristischen Leuchtdichteverteilungen und die daraus resultierenden Einflüsse für ein effizientes Local-Dimming System wichtig sind. Neben der unterschiedlichen Dimming Zonen Anzahl können sich diese charakteristischen LSFs sehr stark unterscheiden. So gibt es sowohl radial-symmetrische als auch rechteckige LSF Formen. Im nächsten Abschnitt wird auf weitere Bewertungsmöglichkeiten von BLUs eingegangen.

## 4.2. Qualitätsmetriken für hochauflösende BLUs

Im Automobil hat die Minimierung der Leistungsaufnahme einen hohen Stellenwert, um den Kohlenstoffdioxid Ausstoß zu verringern. Die Leistungseinsparung ist ein wichtiges Kriterium bei der Bewertung von Displays

## 4.2. Qualitätsmetriken für hochauflösende BLUs

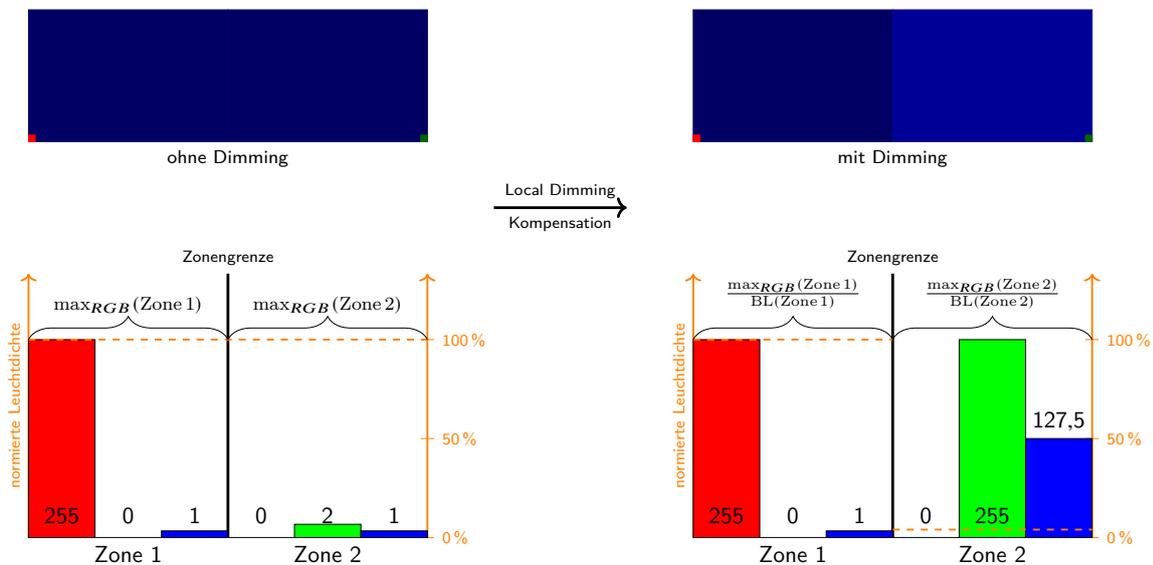


Abbildung 4.24.: Visualisierung der Entstehung von sichtbaren Blöcken durch zu steile LSFs.

(LC-Panel, Backlight und Ansteuerelektronik). Den höchsten Strombedarf hat die BLU, welcher jedoch durch Local-Dimming Algorithmen reduziert werden kann. Deren Effizienz kann dann durch die Messung der Reduktion der Leistungsaufnahme angegeben werden. Mit dieser Kennzahl wird rein der Algorithmus im Zusammenspiel mit den Eigenschaften der BLU in Bezug auf die Leistungsaufnahme charakterisiert. Dazu hat die IEC ein genormtes Testvideo und Messverfahren zur Bestimmung der Leistungsaufnahme entworfen [DIN]. Das Video zeigt verschiedene Szenen des typischen Inhalts auf einem TV-Gerät. Beispielsweise Nachrichten, Filmausschnitte und weitere. Folglich ist das Video wegen ihres Inhalts nicht für das ICD im Automobil geeignet. In Kapitel Abschnitt 3.1 wurde auf die Besonderheiten von HMI-Inhalt näher erläutert und die Unterschiede aufgezeigt.

Die reine reduzierte Leistungsaufnahme durch den Algorithmus reicht jedoch nicht, um die Wertigkeit des gesamten LCD abzubilden. Die Wertigkeit bzw. das Gefühl des Fahrers, vor einem hochwertigen Display zu sitzen, ist insbesondere für Premium-OEMs wichtig. Beispielsweise wird durch ungleichmäßige Leuchtdichteverteilungen die Qualität für den Betrachter vermindert. Um die Wertigkeit des Displays zu erhöhen, bedarf es nicht nur eines guten Local-Dimming Algorithmus, welcher nur in einem Mindestmaß von der Soll-Leuchtdichte abweicht und das *Clipping* minimal ist. Es müssen auch Eigenschaften des LC-Panels und der BLU für die Qualitätsbewertung berücksichtigt werden. Eigenschaften der BLU wirken sich nicht nur auf das Ergebnis des Algorithmus, sondern auch auf die Homogenität (oder auch Uniformität) des resultierenden Bildes aus. Die entsprechenden Faktoren sind unter anderem die Anzahl der LEDs, dessen Abstandes und die LSF-Form (inklusive der Kavitäten). Da Matrix-BLUs in Automobilen neu sind, wird eine Methode zur Quantifizierung der lokalen Homogenität im nächsten Abschnitt (4.2.1) eingeführt.

Zudem bestimmen die Eigenschaften des LC-Panels, wie beispielsweise die Schwarzwerte, die Ästhetik der anzuzeigenden Bilder ebenfalls erheblich und beeinflussen zusammen mit der BLU und Local-Dimming Algorithmus das Ausmaß des *Halo* Artefaktes (siehe Abschnitt 6.2). Folglich müssen diesen Eigenschaften ebenfalls bei der Qualitätsbestimmung berücksichtigt werden. Dazu wird eine Methode in Abschnitt 4.2.2 eingeführt.

### 4.2.1. Quantifizierung der Homogenität

Bereits in Abbildung 4.22 wurde ersichtlich, dass je nach BLU Design, also der LSF und der Anzahl an LEDs, die Hintergrundbeleuchtung durchaus inhomogen erscheinen kann, wie es bei der 133er BLU der Fall ist. In der genannten Abbildung ist eine Änderung der lokalen Leuchtdichte mit einer bestimmten Ortsfrequenz erkennbar und degradiert somit die gefühlte Qualität des gesamten Displays.

Zur Verdeutlichung wird in Abbildung 4.25 der Einfachheit halber, ein horizontaler Querschnitt durch das 133er All-On betrachtet (blau). Diese regelmäßigen Änderungen in der Leuchtdichteamplitude können gut mit trigonometrischen Funktionen  $\sin$  und  $\cos$  simuliert werden. Beispielhaft wird die Funktion

$$I(x) = a \cdot \sin(b \cdot x + \phi) + c$$

zum fitten verwendet.  $I(x)$  wird in der Abbildung gestrichelt dargestellt. Dabei wurden die Parameter anhand eines Ausschnittes des Verlaufs der Messdaten gewonnen. Es ergeben sich daraus folgenden Parameter von  $I(x)$ :

$$a = 14,58$$

$$b = 2\pi \cdot 0,01$$

$$c = 588,67$$

$$\phi = 3,89$$

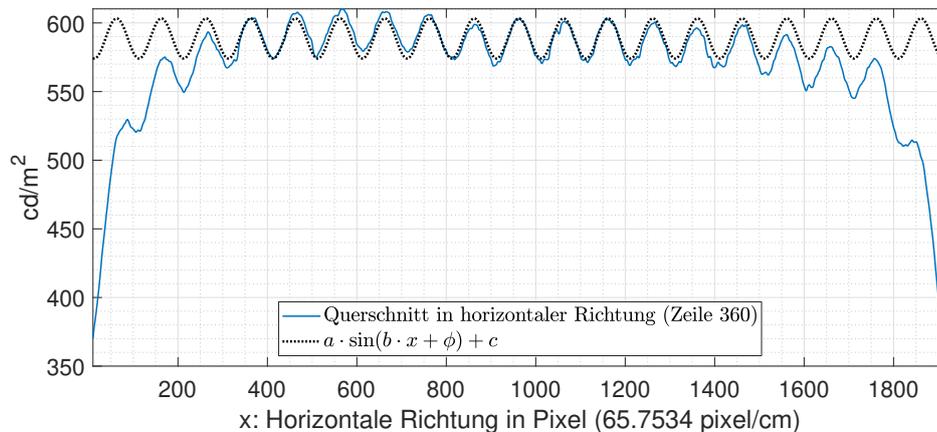


Abbildung 4.25.: Querschnitt durch das absolute All-On der 133 BLU mit Fit einer sinusförmigen Funktion.

Daraus lässt sich direkt der Spitze-Tal-Wert von  $2 \cdot a = 29,16 \text{ cd m}^{-2}$  bestimmen, welcher sich um den Mittelwert von  $588,67 \text{ cd m}^{-2}$  einstellt. Weiterhin ergibt sich eine Frequenz von  $0,01$  pro Pixel. Somit passen komplette 19 Perioden in die horizontale Richtung und entsprechen der LED Zahl. Der maximale Wert beträgt  $603,24 \text{ cd m}^{-2}$  und der minimale  $574,08 \text{ cd m}^{-2}$ . Wird nun ein Abstand ( $d$ ) des Fahrers von dem ICD von  $90 \text{ cm}$  angenommen, so spannt der Winkel  $\alpha_h$  vom Betrachter aus das Display in horizontaler Richtung auf. Generell wird dieser Winkel als *scheinbare Größe* oder *Gesichtswinkel* [Dud] bezeichnet, in dem Beispiel wird jedoch nur die Horizontale betrachtet. Die Breite der Active-Area des ICD beträgt  $29,2 \text{ cm}$  und somit

gilt:

$$\tan\left(\frac{\alpha_h}{2}\right) = \frac{0,5 \cdot 29,2 \text{ cm}}{d}$$

Für die obigen Annahmen entspricht  $\alpha = 18,42^\circ$  und ergibt so eine Ortsfrequenz von 1,03 cpd. Bereits in [CR68] wurde die sogenannte Contrast-Sensitivity Function (CSF, Kontrastempfindlichkeitsfunktion) experimentell ermittelt, dabei wurde für eine mittlere Leuchtdichte von  $500 \text{ cd/m}^2$  die höchste Empfindlichkeit bei 4 cpd festgestellt. Für kleinere Leuchtdichten wandert die höchste Empfindlichkeit zu kleineren Ortsfrequenzen Richtung 1 cpd. Somit wird klar, dass insbesondere für den sehr dunklen Nachtbetrieb dieser BLU, die periodischen Amplitudenunterschiede in der Leuchtdichte das Display in seiner Wertigkeit mindern, da diese als störend wahrgenommen werden. Folglich muss dieser Aspekt beim BLU Design berücksichtigt werden. Auch die resultierenden Ortsfrequenzen können zum Vergleich von BLUs verwendet werden, wenn die entsprechenden Spitze-Tal-Werte und mittlere Leuchtdichte mit berücksichtigt werden.

Im Allgemeinen kann die Homogenität eines Displays mit verschiedenen Verfahren bestimmt werden. Wenn nicht explizit anders erwähnt, bezieht sich der Ausdruck auf die *Weiß Uniformität*, d.h. auf ein All-On mit weißem Bildinhalt. Die Uniformität kann dabei in zwei Typen unterschieden werden, in die *punktueller (Sampled)* bzw. die *Flächen (Area)* Uniformität. Dabei wird die abgetastete Uniformität durch diskrete Einzelmessungen mit einem Spektrometer gemessen, während die Flächenuniformität direkt die gesamte licht-emittierende Fläche des Displays aufnimmt. Beispielsweise kann es mit einem Leuchtdichtemesser wie der Eldim UMaster gemessen werden. Im Information Display Measurement Standard (IDMS) werden verschiedene Berechnungsvorschriften für die abgetastete Uniformität genannt und Berechnung über die Abweichung des maximal gemessenen Wertes empfohlen [Dis12, S. 134 ff.]. In obiger Referenz wird vorgeschlagen 5 Punkte für die abgetastete bzw. punktueller Uniformitätsmessung zu nutzen. Die Uniformität  $U$  berechnet sich mit  $L_{\max/\min}$  als maximal bzw. minimal gemessener Wert der Messpunkte als

$$1 - U = \frac{L_{\max} - L_{\min}}{L_{\max}} \quad (4.9)$$

Der VESA Flat Panel Display Measurements Standard (FPDM) Standard schlägt 5 bzw. 9 Messpunkte vor und nutzt dabei die Berechnung aus Formel (4.9). In der Zwischenzeit wurde der FPDM Standard durch den IDMS ersetzt. Die Standard Panel Working Group (SPWG) schlägt dagegen 13 Messpunkte vor, welche über das Display verteilt sind [Sta05]. Die Berechnung erfolgt dabei ebenfalls nach Formel (4.9). In Abbildung 4.26 sind die vorgeschlagenen Messpunkte des FPDM und SPWG dargestellt. Der schwarze Rahmen zeigt die Grenze der *Active Area*, also dem Bereich, in welchem sich die aktiven Pixel auf dem Display befinden. Dieser Bereich des Displays sorgt für die Modulation der Hintergrundbeleuchtung. Ein weiterer Punkt bei der Durchführung der Messungen ist die Ausrichtung des Aufnahmegerätes. So kann das Gerät entweder so ausgerichtet werden, dass die Aufnahme immer senkrecht zum Display aufgenommen werden, oder aber alle durch einen definierten Beobachtungspunkt gehen. Im letzten Fall wird die Kamera in ihrer Position fixiert und nur im Azimut- bzw. Vertikalwinkel variiert, um einen Beobachter mit fester Position nachzubilden. Somit werden die Messpunkte mit unterschiedlichen Betrachtungswinkel gemessen.

Aus der oben genannten Formel (4.9) geht hervor, dass für die finale Berechnung der Uniformität, genau zwei Werte genutzt werden, unabhängig von der Anzahl der Messpunkte. Somit verbleiben zum Beispiel beim Verfahren der SPWG ca. 84 % der Werte ohne weiteren Einfluss.

Um dieses Verhalten zu verbessern, wird in [DG09] eine weitere Methode vorgeschlagen. Dabei werden die

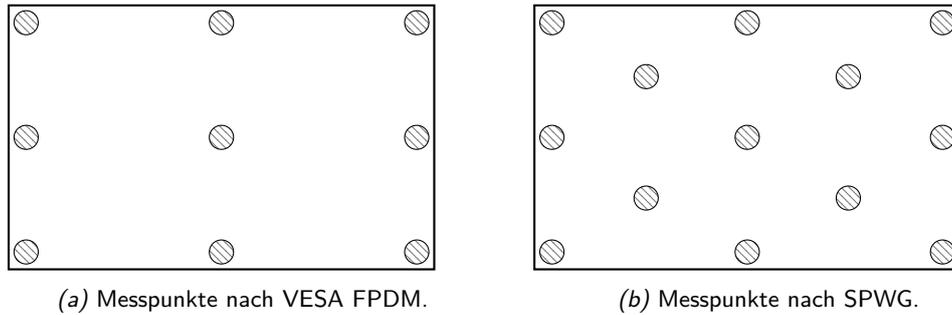


Abbildung 4.26.: Vergleich der vorgeschlagenen Messpunkte auf einem zu vermessenden Display.

Abstände der Messpunkte und deren Leuchtdichtedifferenzen mit in der Berechnung berücksichtigt und entsprechen den Gradienten zwischen den Messpunkten. Für diese Methode wurden die 13 Messpunkte aus SPWG verwendet. Mit  $i, j \in [1, 13]$  und der Abstand zweier Messpunkte  $d_{i,j}$ , definiert sich die Uniformität als:

$$U = 1 - \frac{\sum_{i,j} \frac{|L_i - L_j|}{d_{i,j}}}{\sum_{i,j} \frac{\bar{L}}{d_{i,j}}}, \quad \forall i \neq j \quad (4.10)$$

In [DG09] wird zudem der Abstand  $d_{i,j}$  mit einem Korrektorexponent  $\epsilon$  gewichtet ( $d_{i,j}^\epsilon$ ).  $\bar{L}$  stellt den Mittelwert über die 13 gemessenen Leuchtdichten dar.

Diese Methode beschreibt die Uniformität gut, wenn es sich um die *globale Uniformität* handelt und nutzt im Vergleich zu Formel (4.9) alle Messpunkte unter Berücksichtigung des Messpunktabstandes. Folglich kann diese Methode bei Displays eingesetzt werden, welches systeminhärent nur Änderung der Uniformität mit sehr geringen Ortsfrequenzen zulässt. Also Systeme mit wenigen LEDs und globalem Charakter. Dies ist für Edge-Lit Geräte im Allgemeinen der Fall.

### Lokale Uniformität

Wird Abbildung 4.25 nochmal betrachtet, so ist zu erkennen, dass dort vorhandenen Amplitudenschwankungen mit den 13 SPWG Messpunkten nicht berücksichtigt werden und ein Display nach den obigen Methoden trotz dieser starken lokalen Änderungen als uniform klassifiziert werden. Aus diesem Grund wird eine Methode benötigt, welche es erlaubt, diese hochfrequenten, lokalen Änderungen zu erfassen. Anhand Abbildung 4.25 wird klar, dass die Spitzen mit den Positionen der LED-Zentren korrelieren und motivieren die Betrachtung und Definition einer *lokalen Uniformität* [Sch+18; Sch+19b]. Um diese für Direct-Lit BLUs mit hochauflösender LED-Anordnung zu bestimmen, eignet sich die Idee der Betrachtung der Gradienten bzw. Änderungen der Leuchtdichte pro Distanz der Messpunkte wie in Formel (4.10) nach [DG09] beschrieben. Jedoch müssen charakteristische Messpunkte gewählt werden. Als solche Punkte sind die LED bzw. Kavitäten-Zentren und jeweils die Punkte auf den halben Strecken zwischen diesen geeignet. Folglich sind die zu messenden Punkte von der BLU abhängig, welche vermessen werden soll. Aufgrund der hohen LED-Zahlen wären jedoch zu viele Messungen notwendig. Zum Vereinfachen kann angenommen werden, dass die resultierenden Leuchtdichten einzelner LEDs in ihrer Struktur ähnlich zu den übrigen sind. Dies gilt insbesondere für LEDs, welche sich nicht direkt an den Rändern befinden [Sch+19b]. Mit dieser Annahme können 4 benachbarte LEDs aus dem Zentrum des Panels für die Analyse genutzt werden.

Diese vorgeschlagenen Messpunkte sind in Abbildung 4.27 dargestellt. Die grünen Kreise definieren die einzelnen Messpunkte und die roten die einzelnen LED-Positionen. Insgesamt ergeben sich mit dieser Methode 9 Messpunkte mit Einfluss auf die Berechnung der *lokalen Uniformität*. Anhand dieser Abbildung wird verdeutlicht, dass die Abnahme der Leuchtdichte, am stärksten an der Stelle auftritt, welche den größten Abstand zu den LED-Zentren haben. In diesem Fall also der mittlere Messpunkt. Die Auswertung erfolgt

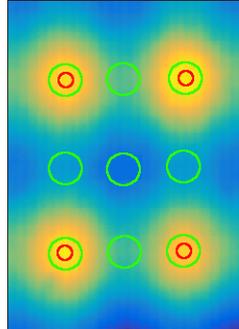


Abbildung 4.27.: Messpunkte zur Berechnung der lokalen Uniformität. Entnommen aus [Sch+18].

dabei nach Formel (4.10) unter Beachtung des Leuchtdichteabfalls zwischen den Messpunkten. Aufgrund der Struktur einer Direct-Lit BLU kann diese Messung repräsentativ für das ganze Display angesehen werden. Als Ergänzung zur globalen Uniformität, welche nach den etablierten Verfahren gemessen werden, liefert sie eine weitere hilfreiche Metrik unter Berücksichtigung von Leuchtdichtegradienten zur Bestimmung der Uniformität und somit auch der wahrgenommenen Wertigkeit des Displays und stellt somit eine sinnvolle Ergänzung zu den Standardverfahren dar, um speziell Direct-Lit BLUs zu bewerten.

### 4.2.2. Schwarze Flächen als Qualitätsmaß

Bisher stütze sich die Wertigkeit eines Direct-Lit Displays auf das All-On ( $L_{\Sigma,t}^{\text{abs}}$  mit  $t = 1$ ). Bei der Betrachtung der Uniformität spielte die licht-modulierende Vorrichtung, also das LC-Panel keine Rolle. Aus Abschnitt 2.4 ist jedoch bekannt, dass die LC-Zellen das Hintergrundlicht nicht auf gänzlich blocken können, sodass auch bei einem Grauwert von 0, eine minimale Leuchtdichte  $> 0 \text{ cd/m}^2$  messbar ist (*Light-Leakage*). Ein hoher Kontrast und gute Schwarzwerte führen zu einer Verbesserung der wahrgenommenen Bildqualität [CH17; HSX16b]. Somit stellt einer der wichtigsten Gründe zur Verwendung von Direct-Lit anstatt von Edge-Lit BLUs dar, die Erwartung den Kontrast des dargestellten Bildes zu verbessern um unter anderem HDR-Anforderungen gerecht zu werden. Realisiert wird diese Verbesserung durch Local-Dimming. Aus diesem Grund wird eine weitere Metrik eingeführt, welche das LC-Panel und indirekt einen Aspekt des genutzten Local-Dimming Algorithmus berücksichtigt.

Besitzt ein Bild viele größere schwarze Flächen, kommen die hellen Pixel gut zur Geltung. Es wirkt hochwertiger als ein Bild mit weniger schwarzen Flächen [Sch+19b]. Das hier vorgestellte Maß beschreibt die Anzahl der *schwarzen* Pixel in einem Bild, welche eine geringere Leuchtdichte haben als  $L_{\text{th}}$ . Dieser Wert kann je nach Anwendungsfall angepasst werden. Wird die VESA *DisplayHDR* Spezifikation als Grundlage für die Parameterwahl herangezogen, kann der Parameter zwischen  $0,0005 \text{ cd/m}^2$  (*DisplayHDR 500 True Black*) und  $0,4 \text{ cd/m}^2$  (*DisplayHDR 400*) liegen [Sto19]. Es sei angemerkt, dass die *True Black* Spezifikationen für OLED-Displays gedacht ist und deswegen nur eine vergleichbar kleine Spitzenleuchtdichte verlangt.

Der höchste Anspruch an ein LCD wird durch *DisplayHDR 1400* vorgegeben mit einem Schwarzwert von  $0,02 \text{ cd m}^{-2}$  und entspricht somit dem 40-fachen Wert. Je nach BLU, Local-Dimming Algorithmus sowie dem Bildinhalt, wird ein schwarzes Pixel einen Schwarzwert und Qualität erzeugen, die ober- oder unterhalb diesem Schwellwert liegt.

Die Formel (4.11) zeigt die Berechnungsvorschrift zur Bestimmung des prozentualen Anteils der schwarzen Pixel im Bild [Sch+19b]. Dabei steht das Tupel  $(i, j)$  für einen Pixel  $p_{i,j} \in \mathcal{P}$ .

$$P_{\text{black}} = \frac{\sum_{i,j} c_{i,j}}{N_P} \text{ mit } \begin{cases} c_{i,j} = 1, L_{(i,j)}^{\text{abs}} \leq L_{\text{th}} \\ c_{i,j} = 0, L_{(i,j)}^{\text{abs}} > L_{\text{th}} \end{cases} \quad (4.11)$$

Diese Kennzahl kann für ein repräsentatives Testset bestimmt werden, sodass beispielsweise ein OEM direkt eine „Panel X ist schöner als Panel Y“ Relation erhält. Die Nutzung dieses Maßes ist insbesondere sinnvoll, wenn das zu vergleichende LC-Panel gleich ist, aber die BLUs variieren. Diese Kennzahl korreliert mit der wahrgenommenen Displayqualität und somit mit dessen Wertigkeit.

## Schlussfolgerungen

In diesem Kapitel wurden verschiedene Methoden gezeigt, um eine hochauflösende BLU hinsichtlich ihrer Qualität zu bewerten. Neben den üblichen Metriken wie *Kontraste* und *Schwarzwerte*, charakterisiert die *Uniformität* und der *Anteil der schwarzen Pixel* (für ein festgelegtes Testset) ein LCD. Zusätzlich wird das *Powersaving* für die Effizienz der Local-Dimming Algorithmus angegeben. Insbesondere der Anteil der schwarzen Pixel sowie das Powersaving wird zusätzlich zu den Eigenschaften der BLU vom Local-Dimming Algorithmus stark beeinflusst. Diese Methoden stellen einem OEM in Kombination gute Metriken zur Bewertung der Wertigkeit eines Displays mit einer hoch aufgelösten BLU zur Verfügung.

### 4.3. LSF Modellierung für Simulations- und Bewertungszwecke

Wie in Abschnitt 4.1.2 bereits geschrieben, kann die Kavität und Optik so gestaltet werden, dass sich eine gewünschte LSF-Form einstellt (radial oder rechteckig). Um nun ein gesamtes Local-Dimming System weiter zu optimieren, muss das BLU-Design zusätzlich zum Algorithmus mit berücksichtigt werden. Da die Form der LSF den größten Einfluss auf die Leistungsfähigkeit und Effizienz des gesamten Systems hat, darf sie beim BLU-Design nicht vernachlässigt werden. Für Abschätzungen und weitere Untersuchungen der LSF-Form bietet sich an, diese Form zu modellieren. Dies geschieht durch eine mathematische Beschreibung. Das Ziel kann die Berechnung der Leistungseinsparung oder auch die Abschätzung der visuellen Qualität sein.

Die Untersuchung der Form auf ihren Einfluss auf Local-Dimming ist essenziell für einen effektiven und robusten Algorithmus und wird in Abschnitt 4.5.1 und 4.5.2 durchgeführt. Hierbei geht es explizit nicht um die Simulation des Lichtweges einer kompletten BLU mittels Raytracing, sondern um die resultierende Leuchtdichte hinter einem Pixel, auch ohne die genaue Kenntnis der optischen Schichten in der BLU selbst. Somit können die Auswirkungen auf Local-Dimming schneller untersucht werden. Im Anschluss kann dann versucht werden, eine als geeignet befundene LSF-Form durch das BLU-Design zu realisieren.

In Abbildung 4.28 ist ein typischer Ablauf des BLU-Designs dargestellt. Hierbei wird die Spezifikation, je nach Position in der Zulieferkette von einem OEM oder Tier-1 vorgegeben. Enthalten sind oft die Anzahl

der Kavitäten und maximale Modultiefe sowie Qualitätsanforderungen wie globale Uniformität. Die weitere Auslegung der BLU geschieht meistens mit Hilfe von Raytracing-Simulationen des gesamten optischen Stacks, jedoch aus Kostengründen mit Einschränkung des Bereiches oder die Anzahl der simulierten Strahlengänge. Nach der simulationsbasierten Optimierung, werden die BLU Prototypen aufgebaut. Im klassischen Ablauf, stellt sich erst an dieser Stelle die Frage, welche Performance bzw. Nutzen ein Local-Dimming System liefert, um die Qualität weiter zu steigern. Die Gründe können vielfältig sein, beispielsweise können die LC-Panels bezüglich ihrem Schwarzwert schlechter als erwartet ausfallen, oder der Strombedarf muss im finalen Produkt deutlich reduziert werden. Unabhängig von den Gründen der *späten* Entscheidung über den Einsatz von Local-Dimming, muss der Einsatz bezüglich Kosten und Flexibilität eine Evaluierung der möglichen Local-Dimming Performance durchgeführt werden. Da in der Regel die nötigen Kompetenzen dazu fehlen, muss die Evaluierung und die Implementierung *extern* durchgeführt werden. Das Ergebnis kann eine IP sein, welche zur finalen Integration ins Gesamtsystem Verwendung findet. Die Herausforderung dabei ist die Tatsache, dass die BLU und somit die LSFs vorgegeben sind. Folglich muss der Local-Dimming Algorithmus angepasst werden, wodurch die Performance nicht im maximal möglichen Bereich liegt, bzw. die Kosten für eine bessere Performance zu hoch wären oder schlicht wegen eines zu globalen Backlights weniger effektiv ist. Gemeinsame Veröffentlichungen mit *Continental* wie beispielsweise [Xu+18b; Sch+16b] zeigen das Interesse der Zulieferer an einer systematischen Optimierung der Dimming-Algorithmen und der BLU. Um bessere Ergebnisse, effizientere Hardware und somit geringere Kosten zu erhalten, wird vorgeschlagen,

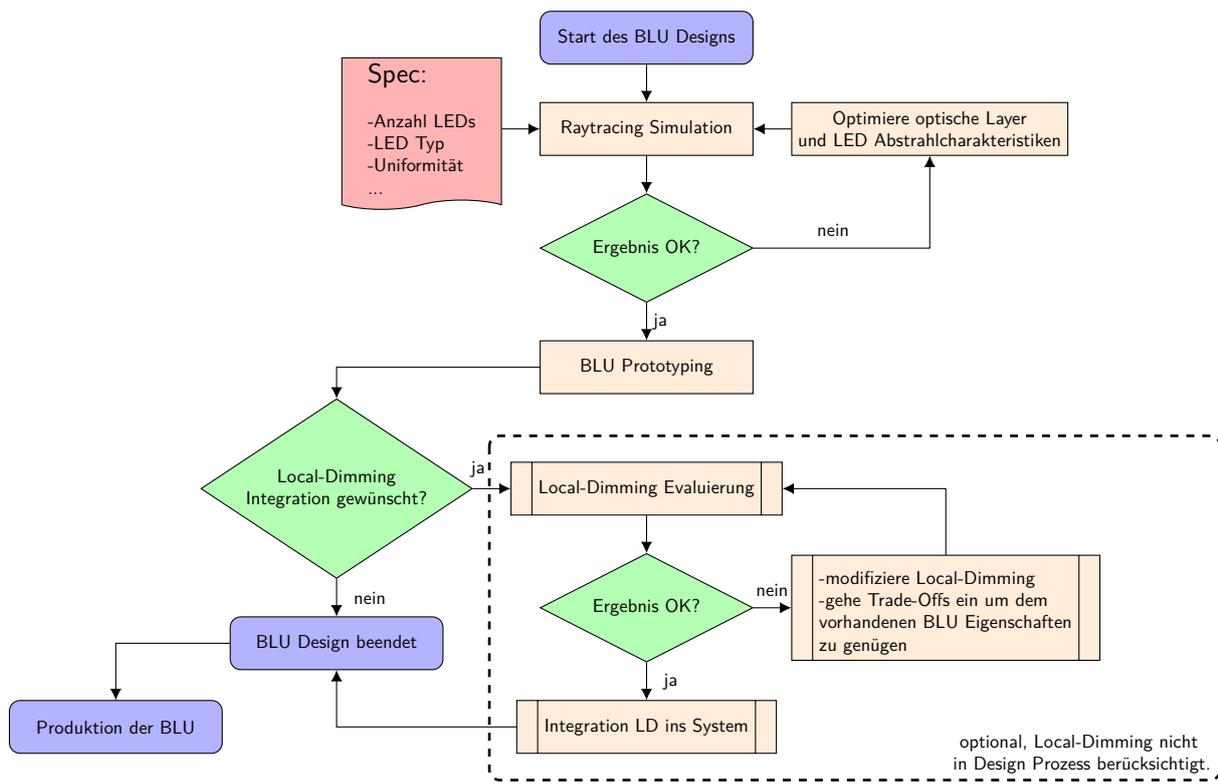


Abbildung 4.28.: Verbreiter Ablauf des BLU-Designs.

dass der Einsatz eines Local-Dimming Algorithmus von Beginn an im BLU-Design Prozess zu berücksichtigen. Dazu ist in Abbildung 4.29 ein Ablauf dargestellt, welcher die wissenschaftlichen Erkenntnisse und Einflüsse der LSF auf den Local-Dimming Algorithmus mit berücksichtigt. Einige der Aspekte wie Lokalität,

Uniformität und Einfluss wurden in diesem Kapitel beschrieben. Zielleuchtdichte, Halo und Ausfall von LEDs werden in Kapitel 6 behandelt. Sie fließen als neue Erkenntnisse in den Design-Prozess mit ein. Ist die Anzahl der Kavitäten nicht vorgegeben, kann eine basierend auf weiterer Spezifikation vorgeschlagen werden. Aus bisherigem Wissen wird eine LSF modelliert, welche als Basis für die Simulation der LSF dient. Diese Simulation kann dann dazu dienen, die Eigenschaften der simulierten BLU zu bestimmen, welche nur von der LSF selbst abhängen sowie Eigenschaften und Performance eines Local-Dimming Algorithmus.

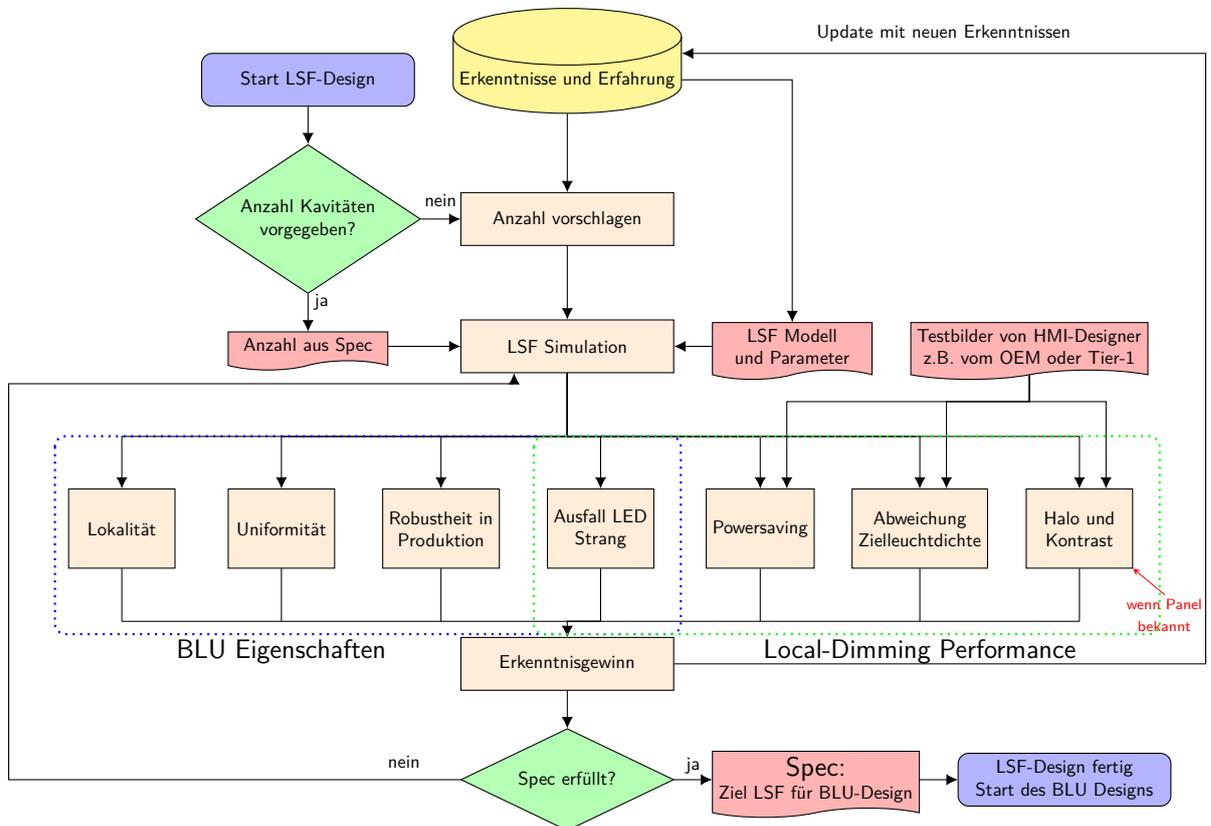


Abbildung 4.29.: Vorgeschlagener Ablauf des BLU-Designs mit Berücksichtigung von Local-Dimming.

Beispielweise ist auch die Methode innerhalb des Local-Dimming Algorithmus zusätzlich zur LSF im Falle eines LED Ausfalles für die Erhaltung der Qualität wichtig. Im Optimum, existieren spezifizierte Testbilder (insbesondere HMI) vom Designer um die Performance verschiedener Local-Dimming Algorithmen bzw. ihrer Einstellungen durch Parameterwahl besser vergleichen zu können. Aus den Resultaten können wiederum neue Erkenntnisse gewonnen werden, welche im Design Prozess berücksichtigt werden können. Sind die Ergebnisse der Simulation gut, so kann der eigentliche, physikalische BLU-Prozess starten. Dabei ist die durchgeführte LSF-Simulation als Ziel bzw. Spezifikation für das zu realisierende Backlight zu verstehen.

### Motivation zur Nutzung eines Modells

Abbildung 4.30 zeigt den Querschnitt durch Leuchtdichtemessungen einer einzelnen LED einer hochauflösender BLU. Dargestellt ist eine Messung mit einer Diffusorfolie und zwei weitere ohne Diffusor, jedoch mit zwei verschiedenen Abständen der Leuchtmittel zum LC-Panel selbst. Diese Abbildungen soll die große

Variationsbreite und Ausprägungen der LSF-Form verdeutlichen. Ebenso wird ersichtlich, dass auch der LED-Abstand zum LC-Panel neben dem Diffusor und weiteren optischen Elementen die Form beeinflusst. Für den genaueren Aufbau der BLU wird auf Abschnitt 2.3.1 verwiesen. Zusätzlich dazu, motiviert das vorgeschlagene Vorgehen aus Abbildung 4.29 die Nutzung eines LSF-Modells um den größtmöglichen Nutzen, also in diesem Fall den Erkenntnisgewinn zu erhalten, mit dem die zukünftige Verbesserung der Qualität und die Reduktion der Hardwarekosten realisiert werden kann.

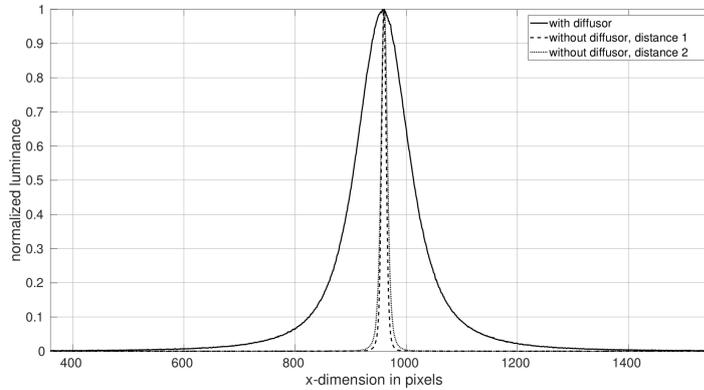


Abbildung 4.30.: Vergleich einer LSF mit und ohne Diffusor. Die Ordinatennachse ist auf die jeweiligen Maxima normiert. Entnommen aus [Sch+19b].

#### 4.3.1. Gauß-Verteilung als LSF-Modell

Ein in der Praxis oft genutztes Modell für die Simulation der LSF ist die Gauß-Verteilung. Beispielsweise in [Tan+18] wird solch eine Verteilung zur Simulation der Leuchtdichte einer Mini-LED BLU genutzt. In [Che+09] wird diese Verteilung ebenfalls verwendet und als Benchmark für die bereits vorgestellten Gauß-Verteilung höherer Ordnung (siehe Formel (4.8) auf Seite 90) eingesetzt. Auch Korhonen u. a. [Kor+11] nutzen die Gauß-Verteilung zum Modellieren eines Local-Dimming Systems basierend auf [AKX08] mit dem Ziel ein durch ein Displayssystem dargestelltes Bild zu simulieren und eine Bildgütebestimmung durchzuführen (Image Quality Assessment (IQA, Bildgütebestimmung)). Die normale zweidimensionale Gauß-Verteilung kann wie folgt beschrieben werden:

$$L_{(i,j),l}^{\text{model}} = A \cdot e^{-\frac{(i-i_0)^2+(j-j_0)^2}{2\sigma^2}} \quad (4.12)$$

Dabei ist  $A$  ein Skalierungsfaktor,  $\sigma^2$  die Varianz und  $(i_0, j_0)$  das Zentrum der Kavität  $l$ .  $(i, j)$  stellen die Pixelkoordinaten eines Pixels  $p_{i,j} \in \mathcal{P}$  dar. In Abbildung 4.31 wird solch ein Modell zusammen mit einer realen Messung im Querschnitt abgebildet. Die Letztere mit einer durchgezogenen Linie dargestellt, das Modell nach Formel (4.12) mit gestrichelten Linien. Es ist direkt zu erkennen, dass die reale Form nicht getroffen wird. Insbesondere der untere Bereich fällt bei diesem Fit zu steil ab und sollte folglich nur für quantitative Simulationen verwendet werden. Dieses Ergebnis kann verbessert werden, indem eine Superposition von Gauß-Verteilungen verwendet wird, um die reale LSF genauer abzubilden. Sei  $N$  die

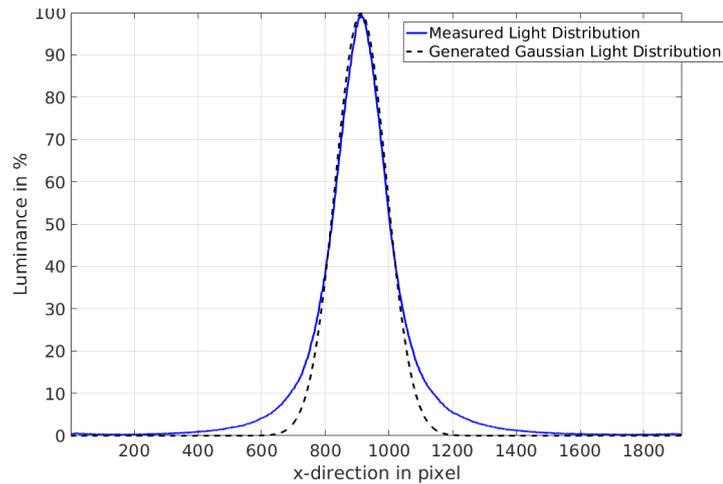


Abbildung 4.31.: Beispiel LSF einer BLU mit 240 Kavitäten und ein einfaches Modell realisiert durch eine Gauß-Verteilung. Entnommen aus [Sch+17].

Anzahl der genutzten Überlagerungen der Gauß-Verteilung, dann definiert sich die Superposition als:

$$L_{(i,j),l}^{\text{model}} = \sum_{n=1}^N A_n \cdot e^{-\frac{(i-i_0)^2 + (j-j_0)^2}{2\sigma_n^2}} \quad (4.13)$$

Der Vorteil der Superposition ist die sehr hohe Genauigkeit, welche für größer werdende  $N$  erreicht werden kann. Der resultierende Nachteil ist jedoch die mit  $N$  steigende Anzahl an Parametern und führt dazu, dass die Parameterwahl schwieriger wird.

Eine weitere Möglichkeit zur Leuchtdichtesimulation wird in [Emm13, S. 19 ff.] vorgestellt und basiert auf der Annahme, dass die LED als ein *Lambert-Strahler* betrachtet werden kann [War00, S. 221 ff.], bietet jedoch den Nachteil, dass auch Parameter wie der LED zu LC-Panel Abstand in das Modell einfließen, dadurch wird jedoch die nötige Abstraktion erschwert, da die physikalische Realisierung durch den angenommenen Abstand einen Freiheitsgrad weniger bekommt. Resultierend aus den Nachteilen der vorgestellten Verfahren, wird im nächsten Abschnitt eine neue Methode eingeführt.

### 4.3.2. Einführung eines neuen Modells für radial-invariante LSF-Formen

Um die Anforderungen an ein gutes Modell zu erfüllen, bedarf es der Einführung einer alternativen Funktion, welche mit möglichst wenig Parametern eine reale LSF nachbilden kann und eine einfachere Analyse und Optimierung erlaubt. Diese neue Modellfunktion wird für radial-invariante, also radial-symmetrische LSFs entworfen, die radial-invariante LSF-Formen von Direct-Lit BLUs immer noch deutlich den Markt dominieren. In diesem Fall gilt die Radialsymmetrie einer 2-dimensionalen LSF repräsentiert als Skalarfeld  $m : \mathbb{R}^2 \rightarrow \mathbb{R}$ , wenn es eine Funktion  $m_d : \mathbb{R} \rightarrow \mathbb{R}$  gibt und für die Funktionswerte des Skalarfeldes  $m$  gilt:

$$m(\mathbf{d}) = m_d(\|\mathbf{d}\|)$$

$\|\mathbf{d}\|$  entspricht dabei dem Abstand vom Ursprung. In Abbildung 4.32 ist ein Querschnitt einer LSF-Messung dargestellt. Die schwarze Linie stellt die Symmetrieachse dar. Rechts von ihr ist der zu modellierende Abschnitt der LSF mit der durchgezogenen Linie abgebildet. Die gestrichelte zeigt den als symmetrischen

angenommenen Teil an. Die rote Markierung zeigt eine kritische Stelle, welche für Local-Dimming nicht vernachlässigt werden darf. Eine Gauß-Funktion würde an dieser Stelle zu einer hohen Abweichung führen (siehe Abbildung 4.31) und ist für eine genaue Analyse des Einflusses auf Local-Dimming weniger geeignet.

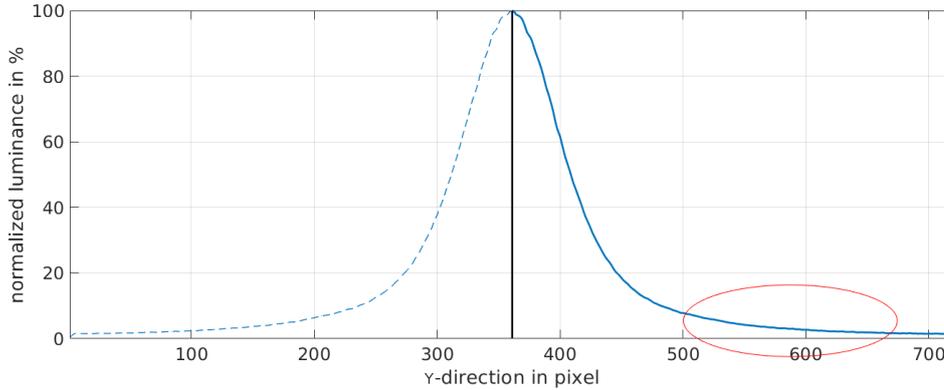


Abbildung 4.32.: Exemplarischer LSF-Querschnitt einer Messung. Mit durchgezogener Linie wird der Bereich dargestellt, welcher modelliert wird. Gestrichelte Linien deuten den symmetrischen Teil an.

Die vorgestellte radial-invariante Modellierungsfunktion  $M : \mathbb{R}_0^+ \rightarrow \mathbb{R}$  setzt sich wie folgt zusammen [SX19]. Die Variable  $x$  repräsentiert in den Formeln den Abstand von der Symmetrieachse:

$$M_1(x) = \frac{1 + e^{-A \cdot B}}{1 + e^{A \cdot (x-B)}} \quad (4.14)$$

$$M_2(x) = x \cdot e^{-\frac{x}{C}} \quad (4.15)$$

$$M(x) = M_1(x) - M_2(x) \cdot \left. \frac{dM_1(x)}{dx} \right|_{x=0} \quad (4.16)$$

Dabei gilt für die Parameter

$$B \in \mathbb{R}_0^+ \wedge A, C \in \mathbb{R}^+$$

und

$$\begin{aligned} \frac{dM_1(x)}{dx} &= - \left(1 + e^{-A \cdot B}\right) \cdot \left(1 + e^{A \cdot (x-B)}\right)^{-2} \cdot A \cdot e^{A \cdot (x-B)} \\ \Rightarrow \left. \frac{dM_1(x)}{dx} \right|_{x=0} &= - \left(1 + e^{-A \cdot B}\right) \cdot \left(1 + e^{-A \cdot B}\right)^{-2} \cdot A \cdot e^{-A \cdot B} \\ &= \frac{-A \cdot e^{-A \cdot B}}{1 + e^{-A \cdot B}} = \frac{-A}{\left(1 + e^{-A \cdot B}\right) \cdot e^{A \cdot B}} = -\frac{A}{e^{A \cdot B} + 1} \end{aligned}$$

Nach der Bestimmung der 3 Parameter wird das LSF-Modell  $M(x)$  um die Symmetrieachse rotiert, sodass ein Skalarfeld

$$L_l^{\text{model}} =: \left( L_{(i,j),l}^{\text{model}} \right), \forall (i,j) \in \{0, \dots, N_P^r - 1\} \times \{0, \dots, N_P^c - 1\}$$

entsteht. Zur Verdeutlichung der Einflüsse der Parameter auf den Funktionsverlauf ist in Abbildung 4.33 eine Funktionenschar von  $M_1(x)$  dargestellt. Im linken Teil der Abbildung ist  $B = \text{const.}$ , im rechten ist  $A$  konstant gewählt. Die Änderungen in der Form zeigen, dass  $A$  großen Einfluss auf den fallenden Teil, also

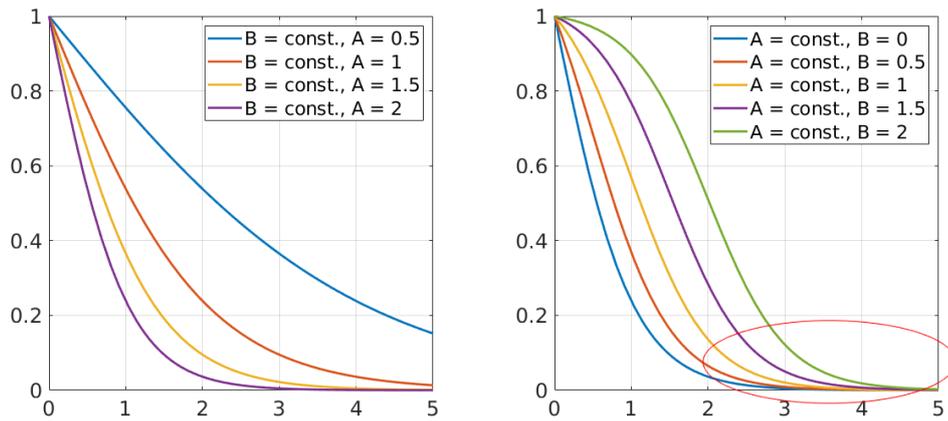


Abbildung 4.33.: Kurvenschar von  $M_1(x)$  mit konstantem  $B$  links und einem konstantem  $A$  in dem rechten Teil (Farbwahl willkürlich).

auf die *Steilheit* der LSF hat, während sich  $B$  gut zur Steuerung der *Breite* eignet.

Außer der Auswirkung der Parametervariation auf die Form, ist außerdem eine weitere Eigenschaft ersichtlich, nämlich der schnelle asymptotische Abfall gegen 0, welcher beispielhaft mit einer roten Ellipse im rechten Bild markiert ist und sich dadurch von einer realen LSF wie aus Abbildung 4.32 deutlich unterscheidet. Dieses Verhalten rührt von den Exponentialfunktionen und entspricht für  $A = -1$  und  $B = 0$  einer Skalierten Sigmoid Funktion  $\text{sig}(x)$ . In diesem Fall läge jedoch  $A$  außerhalb ihrer Definitionsmenge.

Um eine reale LSF gut abbilden zu können, muss das Modell  $M_1(x)$  erweitert werden. Dazu wird  $M_2(x)$  aus Formel (4.15) genutzt. Hier dominiert zunächst der exponentielle Teil. Für große  $x$  ist der lineare Faktor ausschlaggebend.  $M_2(x)$  muss zusätzlich an der Ordinatenachse gespiegelt werden und wird deswegen von  $M_1(x)$  subtrahiert und zusätzlich zur Vermeidung von Unstetigkeit mit  $\frac{dM_1(x)}{dx}$  an  $x = 0$  gewichtet. Visualisiert werden diese Bestandteile in Abbildung 4.34. Dargestellt ist  $M_1(x)$  in blau gestrichelt, welche zu schnell zu geringe Werte annimmt. Rot gestrichelt stellt den gewichteten Subtrahend dar. Die Funktion in orange  $M(x)$  bzw.  $M_{\text{total}}(x)$  im Bild, zeigt den resultierenden Graphen der gesamten Modellfunktion dar. Es ist zu erkennen, dass dadurch  $M_{\text{total}}(x)$  etwas *angehoben* wird, jedoch je nach Parameterwahl nicht so schnell abfällt.

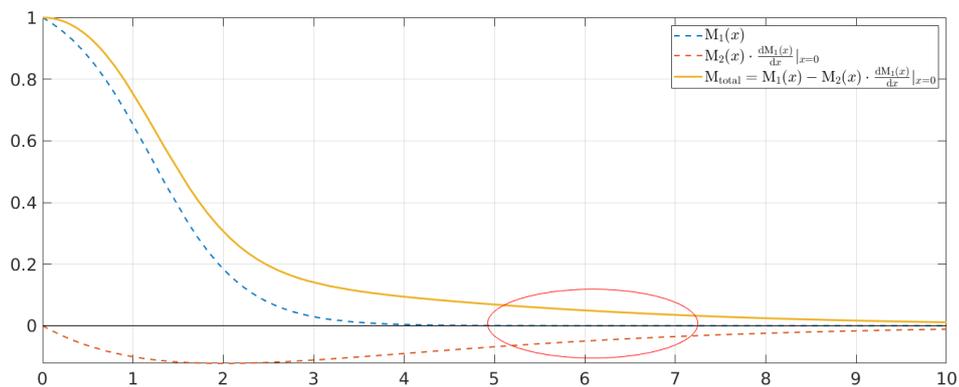


Abbildung 4.34.: Vergleich der Modellbestandteile  $M_1(x)$ ,  $M_2(x)$  und komplette Funktion  $M(x) := M_{\text{total}}(x)$ .

Insgesamt zeigt sich, dass die neue eingeführte Modellfunktion gut geeignet ist, um mit nur 3 Parametern verschiedene LSF Formen generieren und reale LSFs annähern zu können. Sie sind weitgehend unabhängig voneinander und beeinflussen die jeweiligen Bereiche. Der Parameter  $A$  die Steilheit,  $B$  die Breite und  $C$  hat Einfluss auf den unteren Verlauf. Abbildung 4.35 zeigt einen Querschnitt einer realen LSF sowie einem Plot der Models. Es ist gut erkennbar, dass eine sehr gute Näherung möglich ist. Anders als bei der Gauß-Superposition werden hier deutlich weniger Parameter benötigt. Diese Funktion wird in Abschnitt 4.5 genutzt, um den Einfluss verschiedener LSF Formen zu analysieren.

Im nächsten Abschnitt wird ein charakteristischer BLU-Design Parameter eingeführt, der die Local-Dimming Ergebnisse maßgeblich prägt.

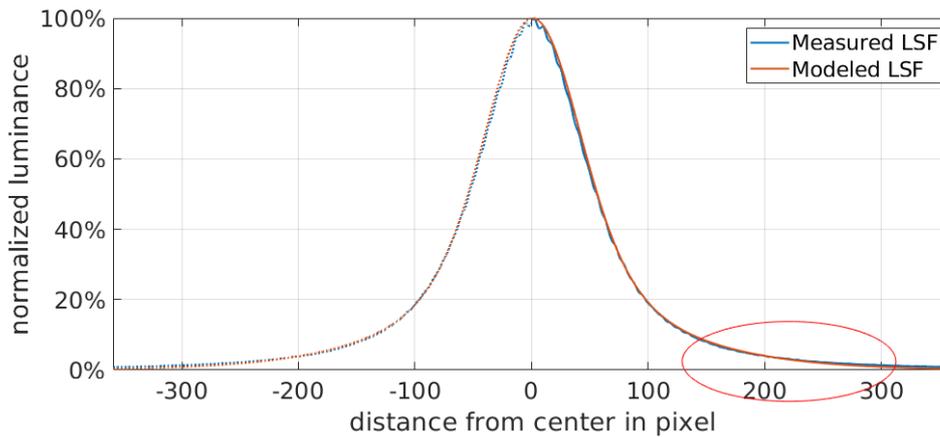


Abbildung 4.35.: Vergleich Modell mit realer LSF. Entnommen aus [SX19].

#### 4.4. Designparameter einer BLU: Das FWHM/Pitch Verhältnis

In vorangegangenen Abschnitten wurden verschiedene LSF Formen betrachtet und Modelle für Analysezwecke vorgestellt und eine Alternative vorgeschlagen. Dabei war nur die Form einer LSF im Mittelpunkt, nicht jedoch das Zusammenwirken vieler LEDs, wie z.B. der Abstand der LEDs zueinander (engl. *pitch*). Als Synonym wird der Begriff LED-Pitch verwendet.

Eine weitere Bemaßung einer LSF ist das FWHM, welche die Weite einer LSF bei 50 % der (normierten) Leuchtdichte angibt. Diese beiden Maße, das FWHM und der LED-Pitch können nun in ein Verhältnis gesetzt werden und als eine mögliche Kenngröße einer BLU angesehen werden. Dieses Verhältnis sei definiert als

$$r_{50\%,x}(l) = \frac{\text{FWHM}(l)}{|\text{pos}(\text{LED}_l)_x - \text{pos}(\text{LED}_{l+1})_x|}, \forall l \in \{0, \dots, N_L - 1\} \quad (4.17)$$

Die Verhältnisse in  $y$  Richtung werden analog kalkuliert. Sind alle LED-Abstände in horizontaler und vertikaler Richtung äquidistant und die LSF Formen idealisiert, können diese Kenngrößen in einem Wert repräsentiert werden. Dabei werden ohne Beschränkung der Allgemeinheit die Werte zweier LEDs in zentraler Lage genommen und das resultierende Verhältnis wird repräsentativ für die BLU angegeben als

$$r_{50\%} = \frac{\text{FWHM}}{\text{Pitch}} \quad (4.18)$$

Ein Beispiel und Visualisierung der Begriffe ist in Abbildung 4.36 am Beispiel zweier LEDs der bereits gezeigten *Hisense* BLU dargestellt. Dieses Beispiel liefert ein  $r_{50\%}$  von 1,95. Dieser Wert ist erfahrungsgemäß sehr hoch und liefert eine vergleichbar geringe Lokalität und Reduktion des Leistungsumsatzes, da der maximale Einfluss bei etwa 10% liegt (siehe auch Abschnitt 4.1.2). Der Vorteil eines so hohen  $r_{50}$  ist meistens ein sehr uniformes All-On.

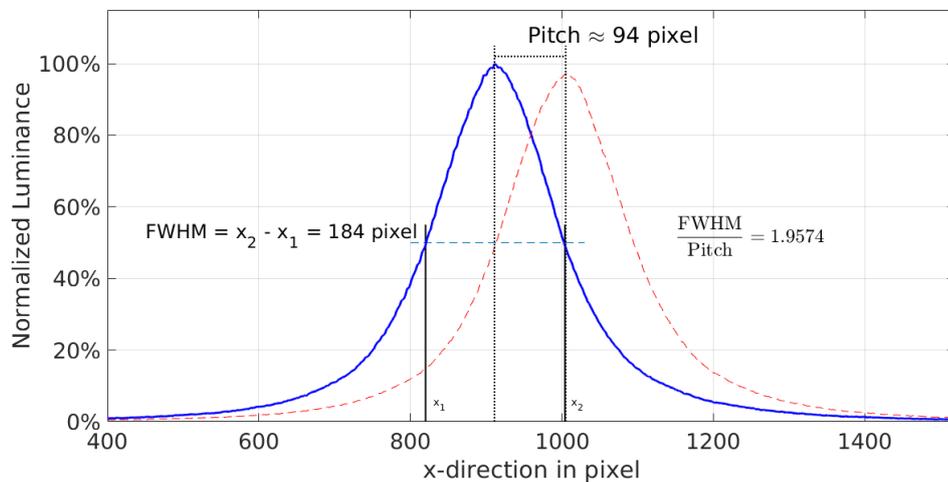


Abbildung 4.36.: FWHM zu LED-Pitch Verhältnis Querschnitte zweier benachbarten LEDs sind dargestellt. Entnommen aus [Sch+17].

Als interessanten Vergleich dient Abbildung 4.37. Diese zeigt jeweils die auf das Maximum normierte Leuchtdichte einen Ausschnitt zweier All-Ons. Dabei die genutzten LEDs und die Dimension der BLU identisch. Sie unterscheiden sich nur durch eine andere Kavitätenanzahl. Dies führt bei dem linken Bild (133er BLU) zu einem  $r_{50\%}$  von 1,1 und bei dem rechten Bild (319er BLU) zu 1,6. So wird klar, dass die visuelle Qualität durch das niedrige  $r_{50\%}$  degradiert wird, wie in Abschnitt 4.2.1 erläutert. Es entsteht eine hohe Ortsfrequenz in der Leuchtdichte. Im Anhang in Abbildungen A.4 bis A.6 sind die FWHM zu

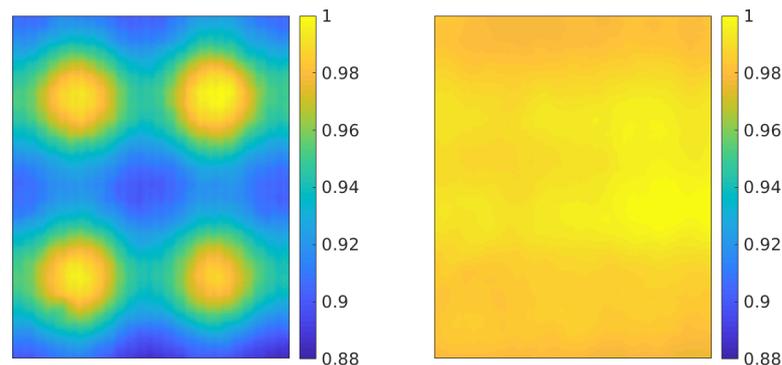


Abbildung 4.37.: Vergleich zweier BLUs mit verschiedenem LED-Pitch und identischer Halbwertsbreiten der LSFs. Linkes Bild  $r_{50\%} = 1,1$  und rechts  $r_{50\%} = 1,6$ . Entnommen aus [Sch+17].

LED-Pitch Verhältnisse der beiden BLUs im Querschnitt dargestellt. Zusätzlich auch das Ergebnis der BLU aus Abbildung 4.19 von Seite 89. Wird nun die LSF von der 133er BLU mit Formel (4.13) angenähert,

kann die Auswirkung des  $r_{50\%}$  weiter verdeutlicht werden. Dazu wird eine weitere BLU mit über 1300 LEDs simuliert. Somit unterscheiden sich die BLUs durch den LED-Pitch. Das FWHM bleibt hingegen identisch. In Abbildung 4.38 ist ein Ausschnitt aus der genannten Simulation dargestellt. Es wird klar, dass mit einem hohen  $r_{50\%}$  bzw. hier mit Verwendung höherer LED Zahlen, das Erreichen einer uniformen Hintergrundbeleuchtung einfacher zu realisieren ist [Sch+19b].

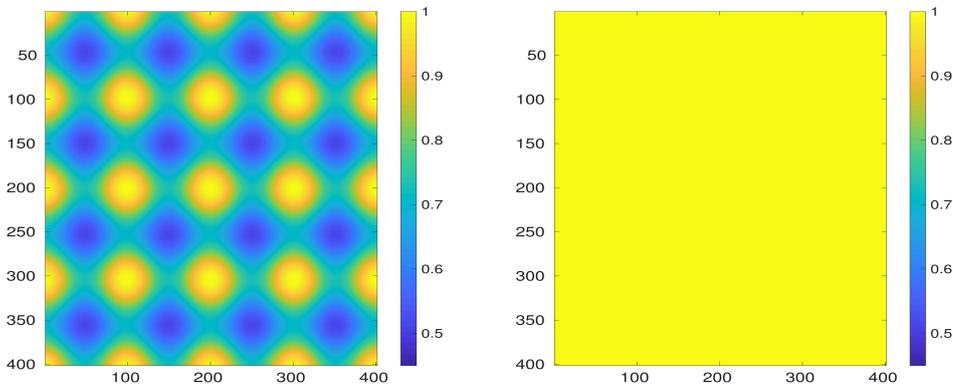


Abbildung 4.38.: Vergleich zweier simulierter BLUs (133 LEDs links vs. 1334 LEDs rechts) mit identischen Halbwertsbreiten der LSFs. Durch den verschiedenen Pitch ergibt sich links ein  $r_{50\%} = 1,1$  und rechts ein  $r_{50\%} = 3,3$ . Entnommen aus [Sch+19b].

Zur weiteren Analyse wird  $r_{50\%}$  gegen den maximalen Einfluss in Abbildung 4.39 visualisiert. Dabei wird das Verhältnis variiert mit  $r_{50\%} \in [1,1; \dots; 2,0]$ . Hier wird ebenfalls das Modell aus Formel (4.13) zur Berechnung der Einflüsse verwendet. Wird das Verhältnis größer, so wird der Einfluss geringer und somit ein Local-Dimming System *globaler*. Ist der Einfluss zu hoch, so kann es leichter zu Artefakten führen und auch eine defekte LED hätte eine stärkere negative Wirkung auf die Bildqualität, da angrenzende LEDs nicht ausreichen, um das Leuchtdichtedefizit zu decken. Basierend auf bisherigen Erkenntnissen, wird ein  $r_{50\%}$  von  $\approx 1,5$  als guter Kompromiss vorgeschlagen [Sch+19b; Sch+17]. Wird nun die Anzahl

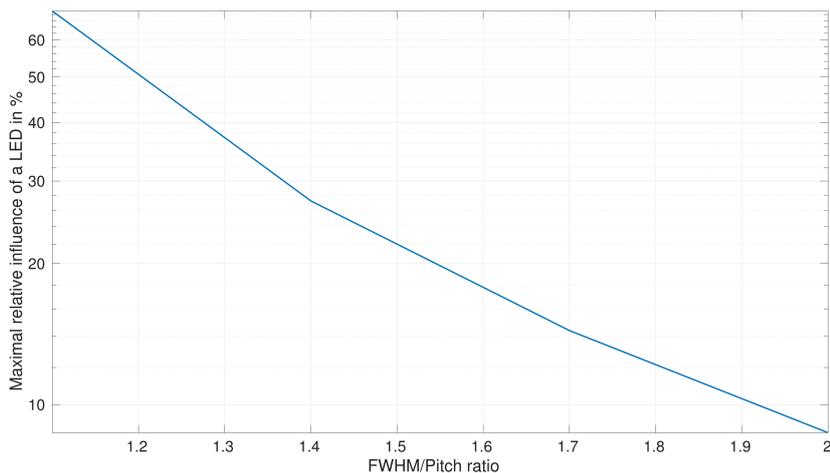


Abbildung 4.39.: Maximaler Einfluss in Abhängigkeit vom FWHM/LED-Pitch Verhältnis. Entnommen aus [Sch+19b].

der LEDs variiert und der mittlere und maximale Einfluss gegen die steigende LED Zahl aufgetragen, so ergibt sich der Plot in Abbildung 4.40. Es wird jeweils ein  $r_{50\%}$  konstant gehalten und die LED-Anzahl variiert (Abszisse). Der resultierende maximale Einfluss ist in Blau gezeichnet (rechte Ordinatenachse), der mittlere Einfluss in Schwarz (linke Ordinatenachse). Für ein  $r_{50\%} = \text{const.}$  ergibt sich unabhängig von der LED-Anzahl derselbe maximale Einfluss. Der mittlere Einfluss hingegen fällt reziprok mit steigender LED-Zahl, ist jedoch unabhängig von  $r_{50\%}$  und gilt für alle Verhältnisse (in Abbildung als „all FWHM/Pitch ratios“ bezeichnet). Diese Abbildung deckt sich mit dem Ergebnis der Herleitung aus Formel (4.6) auf Seite 84 welche zeigte, dass der mittlere Einfluss dem Kehrwert der LED-Anzahl entspricht, also  $\bar{L} = N_L^{-1}$ . Aus Abbildungen 4.39 und 4.40 wird deutlich, dass  $r_{50\%}$  neben der Uniformität (vgl. Abbildungen 4.37 und 4.38) auch zur Einstellung des maximalen Einflusses wichtig ist.

Um die Lokalität zu Erhöhen muss nach Formel (4.7) (Seite 84) das  $\Delta \frac{\bar{L}^{\max}}{L^{\max}}$  minimiert werden. Bei einem

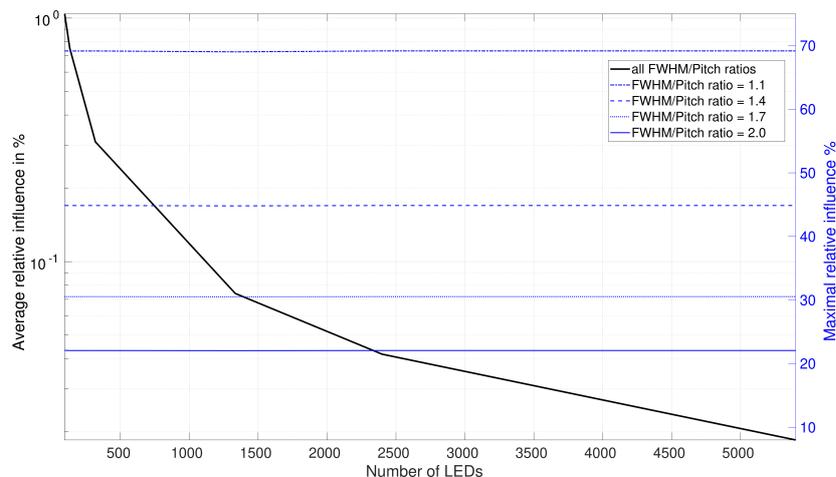


Abbildung 4.40.: Maximale ( $\bar{L}^{\max}_{\text{total}}$ , rechte Ordinatenachse) und mittlere ( $\bar{L}$ , linke Ordinatenachse) Einflüsse in Abhängigkeit von der Anzahl der LEDs. Entnommen aus [Sch+19b].

festen maximalen Einfluss muss der mittlere Einfluss von  $\bar{L}^{\max}$  folglich durch die LSF-Form maximiert werden. Es soll angemerkt werden, dass der mittlere Einfluss von  $\bar{L}^{\max}$  aus Formeln (4.4) bis (4.5) (Beispiel  $\bar{L}^{\max}$  von in Abbildung 4.11 auf Seite 83) nicht zu Verwechseln mit dem in Abbildung 4.40 dargestellten mittleren Einfluss (schwarz) ist (siehe Formel (4.6)).

In Abbildung 4.41 werden simulierte All-On-Bilder für eine konstante LED Anzahl  $N_L = 319$  bei variiertem  $r_{50\%}$  gezeigt. Die verschiedenen Maximalwerte beim All-On rühren daher, dass die Modelle alle auf 100 % normiert sind. Durch ein größeres  $r_{50\%}$  wird also die Superposition höher. Klar zu erkennen ist außerdem die stetig wachsende Uniformität der BLU mit steigendem  $r_{50\%}$ . Unten im Bild sind die Querschnitte durch die Einflüsse dargestellt. Wird nur ein definierter Bereich betrachtet (grün umrandet), so nehmen die Einflüsse mit steigendem  $r_{50\%}$  ab. Der Bereich ist so gewählt, dass die Kantenlänge dem LED-Pitch entspricht. In diesem Abschnitt wurde gezeigt, wie sich FWHM/Pitch Verhältnis auf den Einfluss der LSFs auswirkt. Somit stellt diese Kennzahl eine wichtige Kenngröße des BLU Designs bereit. Im nächsten Abschnitt werden die eingeführten Größen wie  $r_{50\%}$  und Einflüsse verwendet, um die Auswirkungen der LSF Form auf die BLU Qualität zu untersuchen.

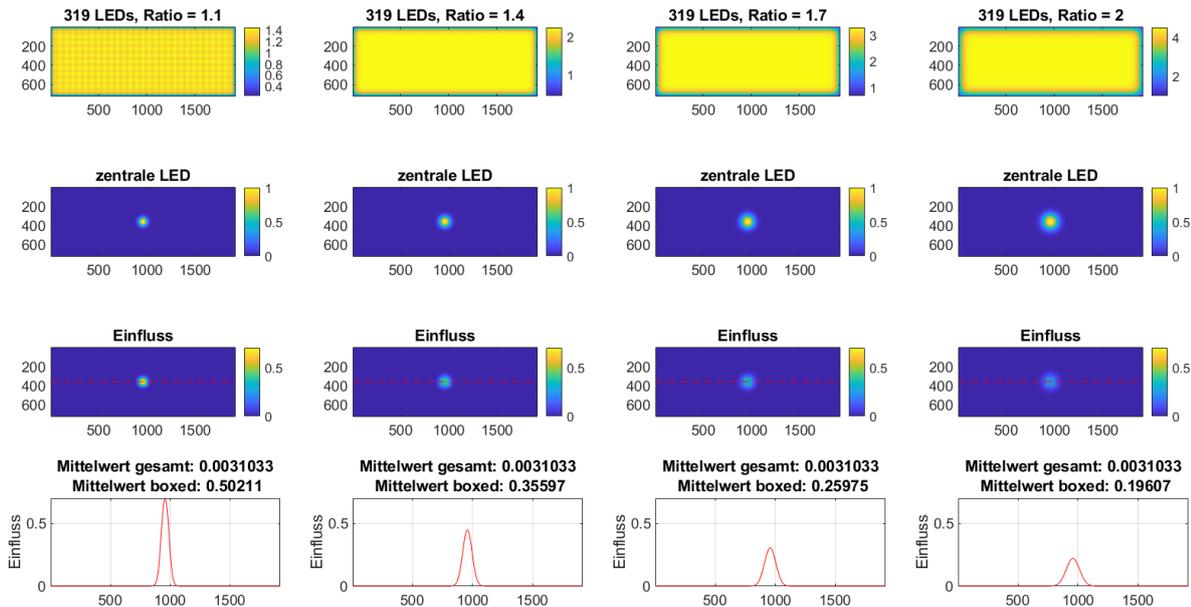


Abbildung 4.41.: Mittlerer Einfluss und All-On bei fester Kavitätenanzahl und variiertem  $r_{50\%}$ .

## 4.5. Auswirkungen der LSF-Form auf die Robustheit und Qualität

In Abschnitt 4.1.2 wurden bereits verschiedene LSF-Formen vorgestellt und gezeigt, dass die Form einen wichtigen Beitrag zur Qualität und Local-Dimming Effizienz leistet. In diesem Abschnitt wird die Auswirkung der LSF-Form auf die *Robustheit* bei der Produktion und auf die Qualität des Backlights untersucht. Dabei bezieht sich der Begriff *Robustheit* auf den Kontext der Fertigung der BLUs und vorhandener Produktionsschwankungen. Beispielsweise können im Fertigungsprozess die LED-Positionen etwas variieren, sowie die LSF-Form an sich leichte Unterschiede aufweisen. Die Simulationen in diesem Kapitel basieren auf dem LSF-Modell, welches in Abschnitt 4.3.2 eingeführt wurde, da wenige Parameter verwendet werden, weswegen die Variation der Parameter nicht zu komplex wird. Die Genauigkeit bzw. Auflösung der simulierten BLUs werden durch die native Panelauflösung des Displays repräsentiert. Beispielsweise sei ein ICD mit einer Auflösung von  $720 \times 1920$  Pixel gewählt, sodass eine simulierte LSF einer LED analog zu Formel (2.14) repräsentiert wird.

Es ist davon auszugehen, dass die in Abschnitt 4.4 eingeführte Kennzahl als alleiniges Kriterium nicht ausreicht, um ein gutes Backlight (in Bezug auf Robustheit) zu erzeugen. Die Analysen in den nächsten Abschnitten dienen dem Erkenntnisgewinn und Optimierungspotential bei den Aspekten der BLU-Designs.

### Generierung der Modelle

Zur Erzeugung der Modelle basierend auf Abschnitt 4.3.2 wird eine zweistufige Methode verwendet. Die Methode wird in Abbildung 4.42 in einer verallgemeinernden Flowchart Darstellung gezeigt. *Zweistufig* bedeutet in diesem Fall den Ansatz, Datensätze zu generieren (erste Stufe) und zu selektieren (zweite Stufe). Zunächst werden die Modellparameter basierend auf dem zugrundeliegenden Definitionsbereich eingegrenzt und die einzelnen Schrittweiten der Parameter festgelegt. Diese Menge an Kombinationen werden in einer Datenstruktur beispielsweise in einem Array  $D$  gespeichert. Je kleiner die Schrittweiten der Parameter sind, desto größer wird die Menge an zu berechnenden Modellen. Um auf eine geeignete Berechnungszeit zu

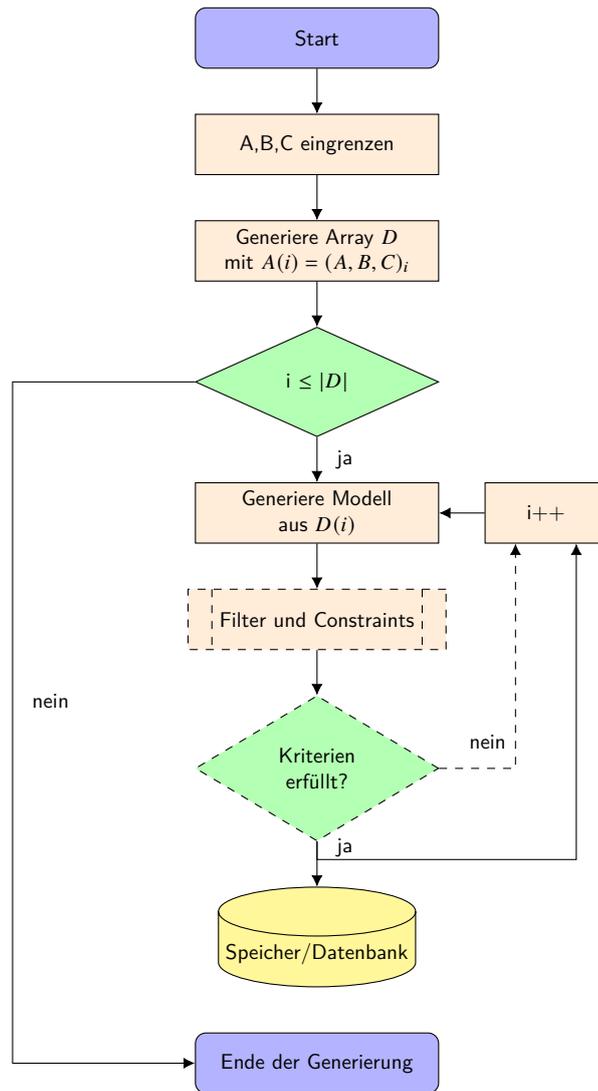


Abbildung 4.42.: Ablauf der LSF-Modell Generierung.

kommen, müssen die Schrittweiten und Grenzen mit Bedacht gewählt werden. Basierend auf  $D$  werden für jedes Parametertripel die Modelle berechnet. Dazu gibt es folgende Möglichkeiten.

### Alternative 1

Zunächst Generierung aller gewählter Parameterkombinationen  $|D|$  und Hinterlegung in einem Speicher. Im zweiten Schritt, nachdem alle Daten vorliegen, können die erzeugten Modelle gefiltert und nach bestimmten Kriterien ausgesucht werden. Der Vorteil ist die Verfügbarkeit aller generierten Modelle in einem Speicher. Der Nachteil ist der je nach  $|D|$  große Speicherbedarf.

### Alternative 2

Bei dieser Möglichkeit werden während der Datengenerierung Filter und Constraints angewandt und nur diejenigen gespeichert, welche die Kriterien erfüllen. Dies können zum Beispiel,  $r_{50\%}$  oder auch die maximale

Steigung sein. Im Flowchart sind die gestrichelten Prozesse in diesem Fall integriert. Der Vorteil ist, dass der Speicherbedarf sehr gering ist, wenn die Filter richtig gesetzt werden, welche mit dem Nachteil einhergeht, dass alle Daten neu generiert werden müssen, wenn die Filter nur minimal geändert werden.

### Alternative 3

Eine weitere Möglichkeit basiert auf einer Mischform der beiden vorangegangenen Möglichkeiten, um die Nachteile der beiden Möglichkeiten zu mindern. Diese Methode nutzt wie auch in der zweiten Möglichkeit, einen Filter im Rahmen der Modellgenerierung, setzt den Filter jedoch nicht so strikt, sodass nur die Modelle ausgefiltert werden, welche mit einer hohen Wahrscheinlichkeit nicht weiter genutzt werden. Die finale Auswahl erfolgt durch weitere Filter und Constraints basierend auf der erzeugten Datenbasis. Somit kann unnötiger Speicher gespart werden, ohne dass bei jeder kleineren Filtereinstellung die Datenbasis neu berechnet werden muss.

Die letztere wird in den folgenden Abschnitten für die Erzeugung von LSF-Modellen genutzt.

#### 4.5.1. Einfluss der LSF-Form bei konstantem FWHM/Pitch Verhältnis

Dieser Abschnitt betrachtet verschiedene LSF-Formen bei einem konstant gehaltenem  $r_{50\%}$ . Wie bereits erwähnt, wird für diese Untersuchung eine radial-invariante LSF angenommen. Für das konstante Verhältnis wird 1,5 gewählt, der wie vorhin beschrieben als optimal angesehen wird [Sch+17; Sch+19b]. Dabei werden die LSF-Formen variiert und der Einfluss auf die Robustheit untersucht. Die Anzahl der Kavitäten wurden in diesem Versuch für eine zukunfts-trächtigen BLU zu  $15 \times 40 = 600$  festgelegt. Die Parameter  $A, B, C$  wurden variiert und 6 verschiedene Tripel ausgewählt, deren LSFs sich stark unterscheiden. Die resultierenden normierten Leuchtdichten sind in Abbildung 4.43a in einem Querschnitt dargestellt und werden in diesem Abschnitt mit

$$\mathbf{L}_i \in [0; 1]^{N_P^r \times N_P^c} \wedge i \in \{1, \dots, 6\}$$

abgekürzt. Es ist auch gut erkennbar, dass die gewählten Parameter sehr verschiedenen Formen realisieren, trotz konstantem  $r_{50\%}$ . Dies wird in dieser Abbildung ebenfalls deutlich. Nach Aussagen von Entwicklern bei BLU-Hersteller, sind solche LSFs realisierbar. Die LED-Positionen sind durch ihre feste Anzahl vorgegeben, so definiert sich  $r_{50\%}$  also nur durch das FWHM. Die resultierenden Parameter sind in der Tabelle 4.2 zu finden. Neben der simulierten Leuchtdichte, ist für ein effizientes Local-Dimming System der Lichtbeitrag, also der

Parameter:	$\mathbf{L}_1$	$\mathbf{L}_2$	$\mathbf{L}_3$	$\mathbf{L}_4$	$\mathbf{L}_5$	$\mathbf{L}_6$
$A$	0,075	0,090	0,075	0,085	0,12	0,275
$B$	26,20	23,60	24,00	18,00	33,20	35,40
$C$	27,20	57,40	18,00	50,00	35,20	76,00

Tabelle 4.2.: Modellparameter für die Generierung der LSFs für konstanten  $r_{50\%}$ .

Einfluss in einem Pixel von Bedeutung. Dieser berechnet sich nach Formel (2.15) und ist in Abbildung 4.43b für die 6 Beispiele abgebildet und unterscheiden sich deutlich. Insbesondere ihr Steigungsverhalten ist sehr unterschiedlich. In Abbildung A.7 auf Seite 245 ist dieses Verhalten in einem Plot dargestellt. Der auffallend hohe Lichtbeitrag von  $\mathbf{L}_5$  und  $\mathbf{L}_6$  kann sehr leicht zu einem höheren Risiko von visuellen Artefakten führen. Die Form von  $\mathbf{L}_4$  beispielsweise führt zu einer schlechten Local-Dimming Performance, da der Einfluss sehr gering ausfällt. Zur Verdeutlichung der Unterschiede, sind die simulierten All-Ons in Abbildung 4.44 im

Vergleich dargestellt und zur besseren Darstellung mit einer Gammafunktion mit  $\gamma = 3$  gestreckt. Dabei fallen die Muster von  $L_5$  und  $L_6$  auf. Jedoch sind die resultierenden Leuchtdichtespitzen nicht etwa die Zentren der LEDs, sondern liegen jeweils zwischen 4 benachbarten LEDs. Ein ähnliches Artefakt wurde auch in [Che+08; LCT07] beobachtet (siehe Abbildung 4.23, Seite 91). Zur Verdeutlichung ist in Abbildung 4.45 das All-on von  $L_6$  mit den ihren LED-Zentren (grün) in Graustufe dargestellt. Erkennbar sind die lokalen Minima im LED-Zentrum. Zwei benachbarte LSFs in vertikaler Richtung werden in Abbildung 4.46 gezeigt.

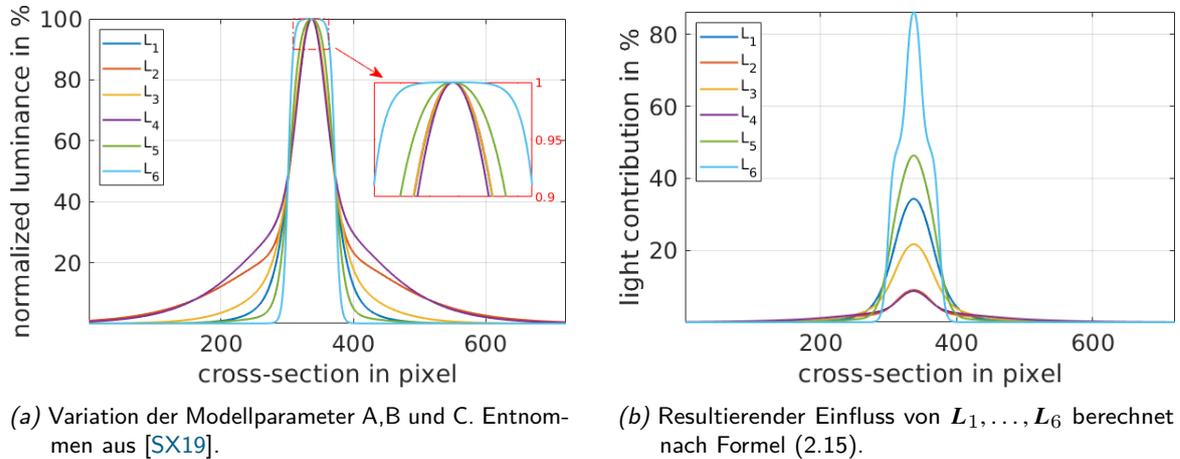


Abbildung 4.43.: Variation der Modellparameter und resultierender Einfluss.

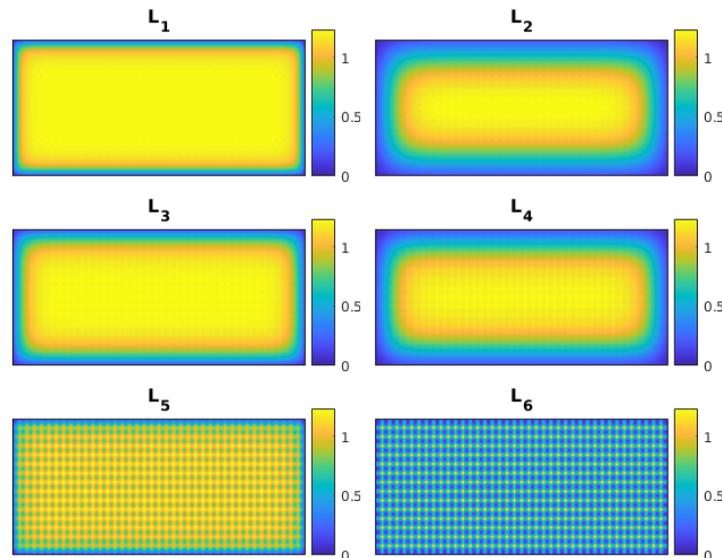


Abbildung 4.44.: Normierte Leuchtdichte simuliert aus obigen Modellen. Zur Visualisierung sind die Daten mit einer  $\gamma$ -Funktion gestreckt. Entnommen aus [SX19].

Dabei ist auch die Superposition der beiden LSFs in Magenta dargestellt. Zu beachten ist außerdem, dass der Maximalwert einer einzelnen LSF hier als 1 gewählt ist. Trotz des gewählten  $r_{50\%}$  von  $\approx 1,5$  überlagern sich die Nachbar LSFs an ihren Grenzen sehr stark. Diese Überlagerung wird durch das breite Plateau um das Zentrum einer LSF verursacht ( $r_{50\%}$  fest), da das Modell erst spät, dafür steil abfällt. Diese Feststellungen

erklären die Beobachtung der auftretenden lokalen Maxima genau zwischen 4 LEDs. Anhand der Differenz zwischen dem gesamten All-On und des Maximums der Superposition der beiden Nachbarn ist erkennbar, dass der Einfluss sehr hoch und nur sehr wenige LED-Nachbarn einen signifikanten Lichtbeitrag leisten.

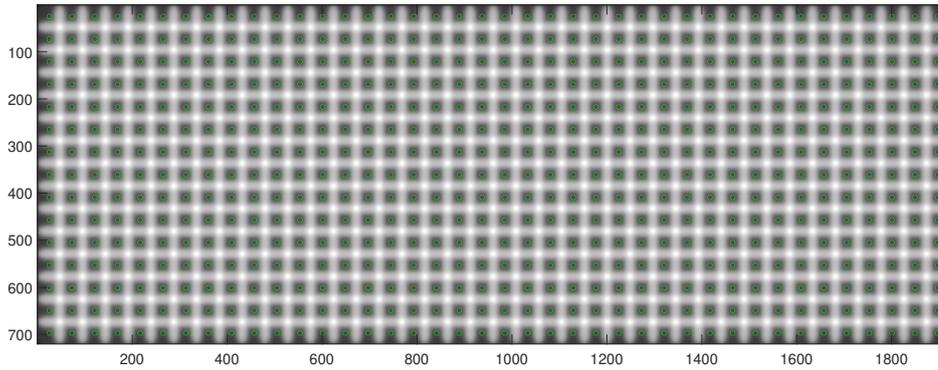


Abbildung 4.45.: All-On von  $L_6$  mit LED-Positionen (grün).

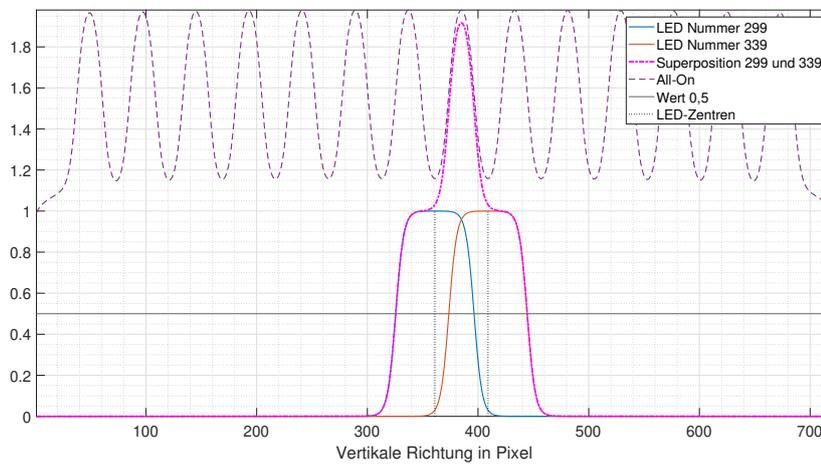


Abbildung 4.46.: Querschnitt durch 2 LSFs des Modells  $L_6$ .

Aufgrund der Form von  $L_4$ , dessen LSF-Zentrum zu spitz ist und ihre Basis für einen *globalen* Charakter sorgt, ist ein leichtes Muster zu erkennen [SX19]. Die Querschnitte durch die kompletten All-Ons sind in Abbildung 4.47 dargestellt und verdeutlichen die Unterschiede noch weiter. Die Skala wurde jeweils auf den höchsten auftretenden Wert normiert. Dass der Maximalwert insbesondere bei  $L_5$  und  $L_6$  nicht erreicht werden, liegt an der Wahl der Lage des Querschnittes. In diesem Fall verläuft der Schnitt durch die LED-Zentren, in welchen sich bei den genannten LSFs lokale Minima befinden (siehe Abbildung 4.45). Folglich gewährt  $L_1$  mit einen maximalen Einfluss von  $\approx 35\%$  eine gute Homogenität und verspricht zugleich ein gutes Local-Dimming Ergebnis.

#### Auswirkungen von inakkurater Platzierung der LEDs

Da in verschiedenen Fertigungsschritten Ungenauigkeiten und Fehler in Bezug auf die exakte LED-Platzierung eingeführt werden können, müssen die Auswirkungen dieser Fehler auf die Qualität der BLU untersucht

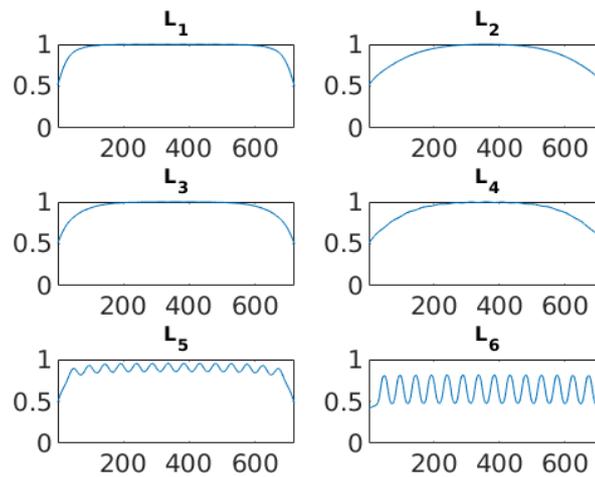


Abbildung 4.47.: Querschnitte durch obige All-Ons der LSF Modelle.

werden. In diesem Abschnitt dient als Qualitätsmaß eine Uniformitätsmetrik. Ein typischer Fehler ist beispielsweise die Fehlplatzierung im Bestückungsprozess der LEDs (Reflow-Löten). Dabei kann sie zum Beispiel in ihrer Ausrichtung leicht variieren. Die Berücksichtigung solcher Fehlermöglichkeiten ist für ein *robustes* BLU-Design wichtig und muss bei der Wahl der LSF-Form vorgenommen werden, um eine zu hohe Empfindlichkeit gegenüber möglicher Schwankungen auszuschließen.

Zum Vergleich der obigen Modelle bezüglich der Auswirkung von inakkurater LED-Platzierung, wird als Referenz die exakte LED-Platzierung angenommen. Ohne Einschränkung der Allgemeinheit werden die Formen von  $L_1$  und  $L_3$  für die Analyse gewählt, da sie sowohl uniformen Charakter haben als auch eine sinnvolle LSF-Form aufweisen. Die Simulation einer Fehlplatzierung erfolgt nun durch zufällige Variation der ursprünglichen LED-Positionen und wird hier auch *LED-Jitter* bezeichnet. Dieser Wert wird als maximal  $\pm 4\%$  des LED-Pitches gewählt und entspricht einem realistischen Verhalten. Als zweiter, zur Visualisierung dienender Wert von  $\pm 20\%$  ist zu hoch gegriffen, erleichtert aber den Unterschied der LSF-Formen zu verdeutlichen [SX19]. Ein Beispiel der Auswirkung solch eines LED-Jitters auf das All-On ist in Abbildung 4.48 dargestellt, dabei wurden für die gewählten LSF-Formen dieselben LED-Positionen mit Jitter zum Zwecke der Vergleichbarkeit verwendet. Die unteren beiden Plots stellen die Querschnitte an den jeweiligen gestrichelten Linien dar. Im Vergleich der Querschnitte der beiden All-Ons lässt sich feststellen, dass die Abweichung vom Optimum (ursprüngliches All-On) bei  $L_1$  deutlich höher ausfällt als bei  $L_3$  und diese Form somit anfälliger auf LED-Jitter reagiert. Die Ursache dieses Unterschiedes befindet sich im Einfluss (Abbildung 4.43b) einer LED basierend auf der LSF. So führt ein geringer maximaler Einfluss zu einem *robusteren* BLU-Design, führt jedoch zu einem tendenziell *globalerem* Verhalten des Local-Dimming Algorithmus.

### Weitere Resultate

In Abschnitt 4.2.1 wurde die Uniformität als Qualitätsmetrik eines BLU eingeführt. Sei im Folgenden  $N_1$  ein Maß für die globale Nicht-Uniformität nach Formel (4.9) und  $N_2$  ein Maß der lokalen Nicht-Uniformität. Die 9 globalen Messpunkte für die Berechnung wurde wie in Abbildung 4.49 gewählt, um den simulierten Leuchtdichteabfall am Rand außen vorzulassen.

Zur Bewertung der Robustheit werden in Abbildung 4.50 für  $L_1$ ,  $L_3$  und  $L_4$  die beiden Metriken  $N_1$  und  $N_2$  berechnet und auf der Ordinate aufgetragen. Die Abszisse zeigt deinen steigenden LED-Jitter als

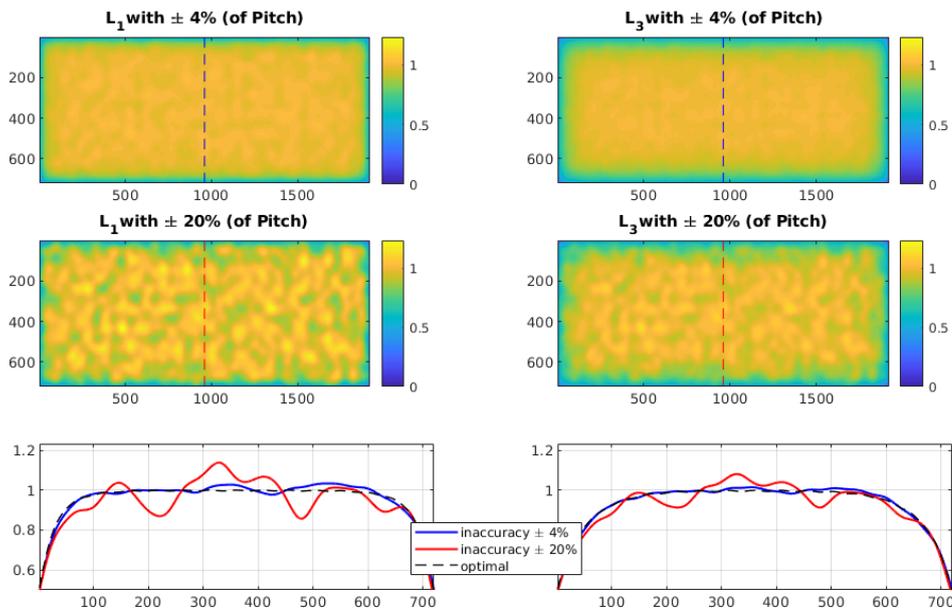


Abbildung 4.48.:  $L_1$  und  $L_3$  unter Annahme eines zufälligen Fehlers bei der LED-Platzierung. Entnommen aus [SX19].

Anteil des LED-Pitches. Erkennbar in diesem Plot ist, dass unabhängig von dem Jitter, das Ergebnis der globalen Uniformitätsberechnung durchgängig ein uniformes Resultat liefert, was jedoch der Wahrnehmung widerspricht (vgl. Abbildung 4.48). Die lokale Berechnung jedoch zeigt einen vielmehr mit der Wahrnehmung korrelierenden Verlauf, so steigt die Nicht-Uniformität für höher werdende Variationen. Nach weiterer Analyse kann untermauert werden, dass ein niedrigerer Einfluss eine BLU robuster gegenüber Schwankungen der LED-Positionierung macht.

Als Schlussfolgerung dieses Abschnittes kann festgestellt werden, dass neben dem  $r_{50\%}$  auch die LSF-Form und auch der relative Lichtbeitrag sowohl Bildqualität als auch die Reduktion der Leistungsaufnahme beeinflusst. Wird nun der Local-Dimming Algorithmus, welcher in Abschnitt 3.5.1 vorgestellt wurde auf ein HMI-Testset (Anhang B) angewandt, so ergibt sich das in Tabelle 4.3 eingetragene Powersaving.

Unter Beachtung von Abbildung 4.43b ist ein Zusammenhang zwischen dem Powersaving und dem maximalen Einfluss (Tabelle 4.3) erkennbar. Der lineare Zusammenhang lässt sich beispielsweise mit dem *Pearson-Korrelationskoeffizienten*  $r$  mit  $r \in [-1; 1]$  beschreiben [Ben+09]. Dieser berechnet sich zu  $r = 0,8725$  und stellt damit mit einem  $p$ -Wert von 0,0233 eine statistisch signifikante positive Korrelation dar. Somit hängt die Local-Dimming Performance sehr stark von dem Lichtbeitrag einer LED ab.

Zum Aufzeigen der verschiedenen Charakteristik der LSFs trotz gleichem  $r_{50\%}$ , sind in Abbildung 4.51 die

	$L_1$	$L_2$	$L_3$	$L_4$	$L_5$	$L_6$
max. Einfluss	34,3 %	8,9 %	21,7 %	8,6 %	46,4 %	86,3 %
Einsparung	50,7 %	40,2 %	48,0 %	41,5 %	51,1 %	54,6 %

Tabelle 4.3.: Maximaler Einfluss der 6 ausgewählten LSF-Formen mit konstantem  $r_{50\%}$  und resultierende Leistungseinsparung.

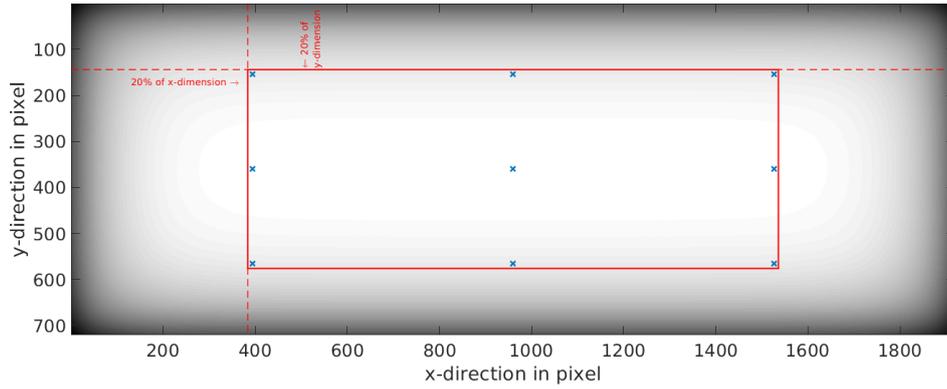


Abbildung 4.49.: Genutzte Messpunkte zur Bestimmung der globalen Uniformität.

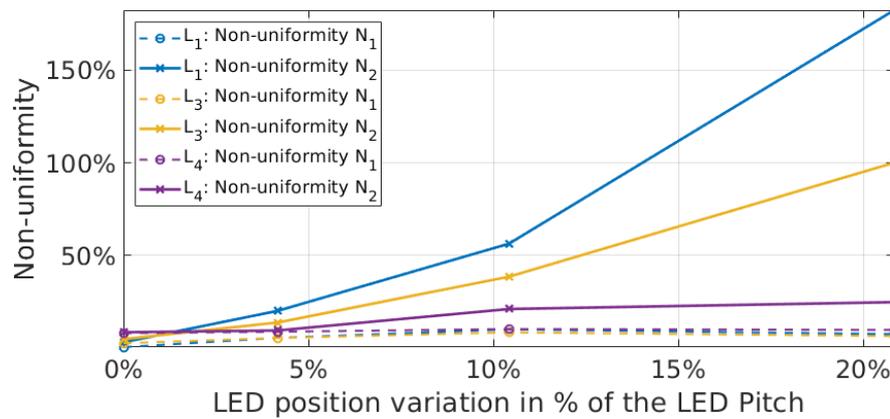


Abbildung 4.50.: Nicht-Uniformitäten bei steigendem LED-Jitter. Entnommen aus [SX19].

resultierenden normierten Leuchtdichten basierend auf den Resultaten des Local-Dimming Algorithmus visualisiert. Die Berechnung des Backlights erfolgt analog zu Abschnitt 3.5.1 und 3.5.2. Mithilfe der modellierten Leuchtdichte, hier allgemein als  $\mathbf{L}^{\text{model}} \in [0; 1]^{N_P \times N_L}$  dargestellt und den berechneten LED-Werten  $\mathbf{v} \in [0; 1]^{N_L \times 1}$  kann das resultierende Backlight wie folgt mit einem Matrix-Vektor-Produkt berechnet werden:

$$\mathbf{L}_{\Sigma}^{\text{model}} = \mathbf{L}^{\text{model}} \cdot \mathbf{v}, \text{ mit } \mathbf{L}_{\Sigma}^{\text{model}} \in [0; 1]^{N_P \times 1} \quad (4.19)$$

Die Darstellung mit linearem Index wird zur Visualisierung in eine Koordinaten basierende Indizierung umgewandelt und als Skalarfeld interpretiert. In der Abbildung sind die Unterschiede der Leuchtdichteverteilung sehr deutlich zu sehen. Die Formen  $\mathbf{L}_2$  und  $\mathbf{L}_4$  haben eine große Ähnlichkeit. Beide sind für diese hohe LED-Anzahl sehr global, was auf den geringen Einfluss zurückzuführen ist. Jedoch ist bei  $\mathbf{L}_4$  ein leichtes Muster der Gitterstruktur sichtbar. Durch das höhere Powersaving von  $\mathbf{L}_1$  gegenüber  $\mathbf{L}_3$  und des höheren Kontrastes, kann insgesamt als Kompromiss eine LSF-Form wie  $\mathbf{L}_1$  vorgeschlagen werden, wenn der LED-Jitter im Fertigungsprozess eingengt werden kann. Es gilt diesen beim BLU-Design neben der Local-Dimming Performance zu berücksichtigen.

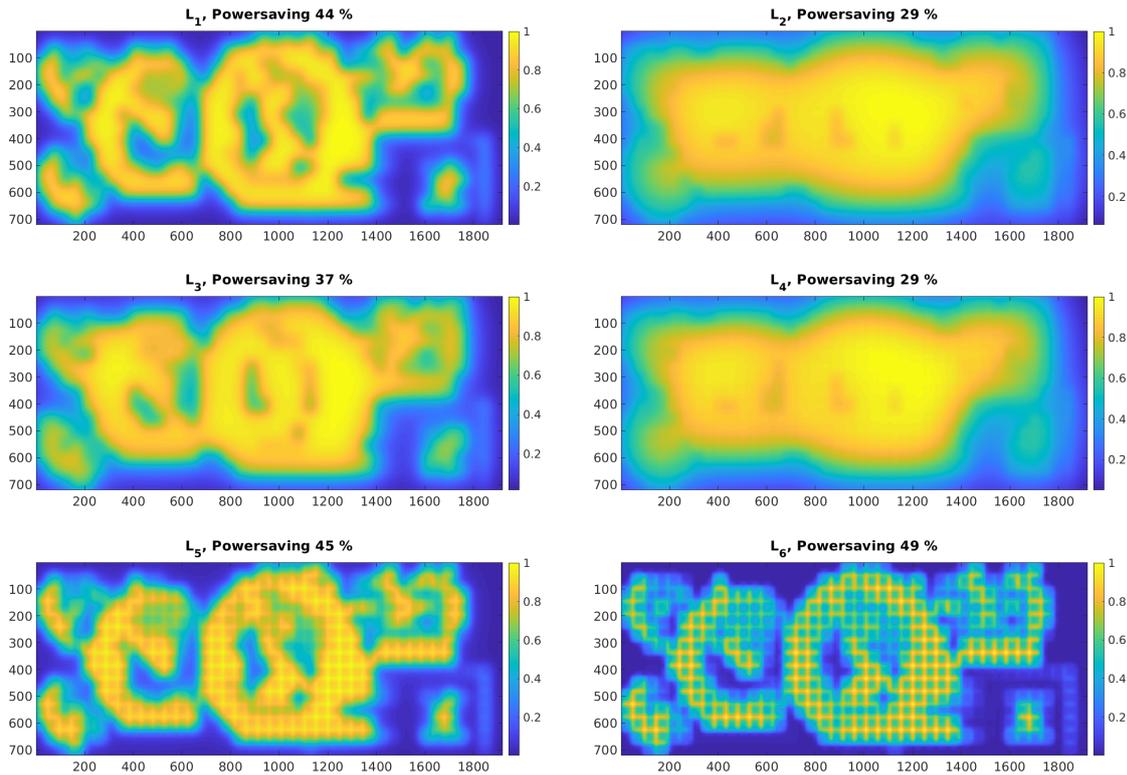


Abbildung 4.51.: Local-Dimming: resultierendes Backlight zum Testbild Nummer 3.

### 4.5.2. Einfluss der LSF-Form mit konstantem maximalen Einfluss

Der vorherige Abschnitt befasste sich mit der Robustheit verschiedener LSF-Formen gegenüber LED-Jitter um die optimalen LED-Positionen in der Backlight-Unit sowie mit den Auswirkungen auf das Backlight sowie Local-Dimming Performance bzw. Eignung. Dazu wurde  $r_{50\%}$  konstant gehalten.

In diesem Abschnitt wird die LSF-Form weiter untersucht. Dabei wird, im Gegensatz zu Abschnitt 4.5.1, der maximale Einfluss konstant gehalten, während  $r_{50\%}$  variieren kann. Dieser konstante Einfluss soll einen Wert von 30% haben. Die Modelle werden ebenfalls mit der Funktion aus Abschnitt 4.3.2 generiert. Aufgrund von hohem Interesse für eine kostengünstige BLU seitens der OEMs wurde die Anzahl der LEDs ( $N_L$ ) auf  $9 \times 24 = 216$  für diese Untersuchung festgelegt. Dieser maximale Einfluss befindet sich bei radial-invarianten LSFs im Allgemeinen über dem Zentrum einer Kavität. Wird nun eine LED mit den LED-Zeilen bzw. Spalten-Nummern  $r$  bzw.  $c$ , wie in Abbildung 4.3 dargestellt, als  $LED_{r,c}$  beschrieben und besitze den linearen Index  $l$ , so liefert die Funktion  $pos()$  die Pixelkoordinaten  $(i_0, j_0)$  (Mittelpunkt) für die gegebenen LED-Zeilen und Spalten. Dann gilt:

$$(i_{0,l}, j_{0,l}) = pos(LED_{r,c})$$

Daraus wird der Constraint formuliert [SRX19]:

$$\check{L}_{(i_0, j_0), l} = \frac{L_l^{\text{model}}(i_{0,l}, j_{0,l})}{\sum_{n=0}^{N_L-1} L_n^{\text{model}}(i_{0,n}, j_{0,n})} \stackrel{!}{=} 0,3 \quad \forall l \in \{0, \dots, N_L - 1\}$$

Diese Bedingung ist jedoch nicht erfüllbar, wenn für jede LED Position die LSFs identische Formen annehmen sollen, da der Einfluss insbesondere an den Rändern höher wird, da die Anzahl direkter Nachbarn sinkt. Um den Einfluss konstant zu halten, müssten sich die LSF-Formen verändern und jeweils eigenständig modelliert werden. Aus diesem Grund soll diese Bedingung nur für die zentralen LEDs der BLU gelten, deren Koordinate mit als  $(r_c, c_c)$  abgekürzt, so gilt  $\text{pos}(\text{LED}_{(r_c, c_c)}) = (i_c, j_c)$ . Dabei ist  $l_c$  der Index einer zentralen LED  $l$ .

$$\check{L}_{(i_c, j_c), l_c} = \frac{\mathbf{L}_l^{\text{model}}(i_c, j_c)}{\sum_{n=0}^{N_L-1} \mathbf{L}_n^{\text{model}}(i_c, n, j_c, n)} \stackrel{!}{=} 0,3 \quad (4.20)$$

Auch in dieser Simulation wurden 6 verschiedene Modelle aus einem generierten Pool zum Vergleich ausgewählt und werden in diesem Abschnitt mit  $\mathbf{M}_1 \dots \mathbf{M}_6$  bezeichnet. Abbildung 4.52 zeigt die 6 Modelle und visualisiert die Unterschiede in der Vergrößerung. Die entsprechenden Einflüsse sind in Abbildung 4.53

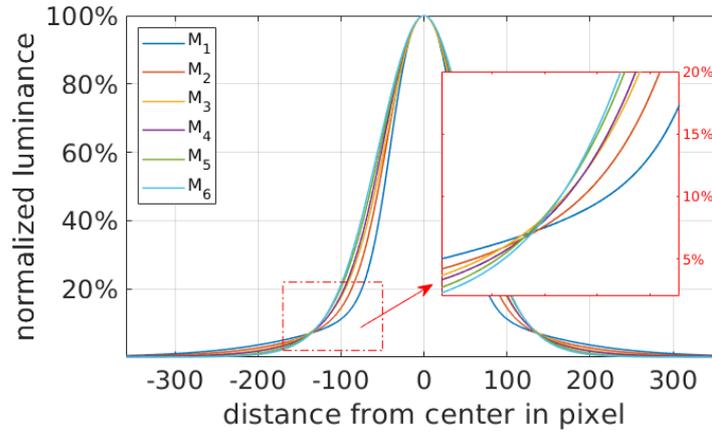


Abbildung 4.52.: Variation der Parameter A, B und C für eine 216er BLU bei konstantem maximalen Einfluss. Sortiert nach steigendem FWHM. Entnommen aus [SRX19].

abgebildet und sind wegen des konstanten maximalen Einflusses (30%) ähnlich zu den Formen der Leuchtdichtemodelle. In Abbildung 4.54 werden die simulierten All-Ons dargestellt und lassen einen Unterschied in der Uniformität zwischen den einzelnen Modellen erkennen. Die resultierenden Parameter sind in Tabelle 4.4 zu finden.

Parameter:	$\mathbf{M}_1$	$\mathbf{M}_2$	$\mathbf{M}_3$	$\mathbf{M}_4$	$\mathbf{M}_5$	$\mathbf{M}_6$
A	0,065	0,050	0,040	0,040	0,035	0,035
B	39,50	42,00	38,00	44,50	47,50	55,50
C	61,50	54,00	46,00	46,50	38,00	29,00

Tabelle 4.4.: Modellparameter für die Generierung der LSFs für konstanten maximalen Einfluss.

Die Auswahl der obigen 6 Modelle wurde vereinfacht, indem zusätzliche Constraints bei der Selektion genutzt wurden. In diesem Fall sind zwei weitere Uniformitätsmetriken ( $U_1, U_2$ ) definiert, welche wie in Abbildung 4.42 beschrieben, als zusätzlicher Filter während der Modellgenerierung verwendet werden können. Wird das All-On als  $\mathbf{L}_{\Sigma, \text{all}}$  bezeichnet (analog zu Abschnitt 4.1.1), dann definiert sich das Maß  $U_1$  als

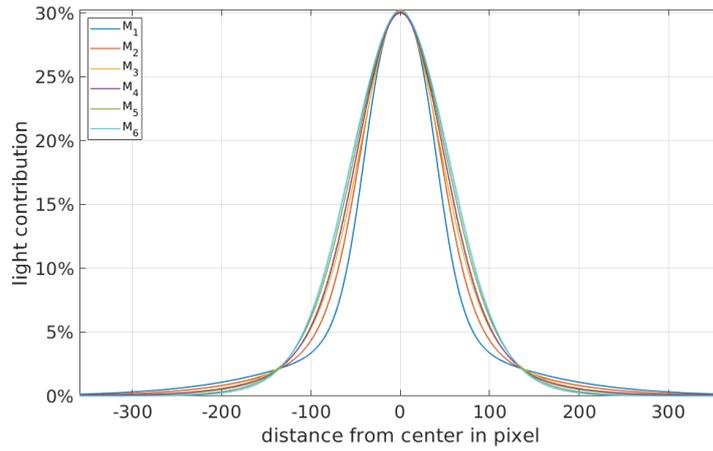


Abbildung 4.53.: Resultierender Einfluss von  $M_1 \dots M_6$  berechnet nach Formel (2.15) (Seite 33).

(analog zu [SRX19]):

$$U_1 = \frac{\sum_{i=0}^{N_P^r-1} \sum_{j=0}^{N_P^c-1} L_{\Sigma, \text{all}}(i, j)}{N_P^r \cdot N_P^c \cdot L_{\Sigma, \text{all}}(\text{pos}(\text{LED}_{(r_c, c_c)}))} \quad (4.21)$$

Mit  $U_1$  wird somit ein Maß für die Abweichung der gemittelten simulierten Leuchtdichte der BLU von der Leuchtdichte über einer im Zentrum liegenden LED gegeben und liefert eine schnell zu berechnende Metrik für die Modellselektion. Das zweite Maß  $U_2$  basiert auf der größeren 2-Norm der diagonalen imaginären Verbindungslinie zweier LEDs, verglichen mit dem horizontalen bzw. vertikalen Abstand. Dieser höhere (euklidische) Abstand impliziert auch die betragsgrößten lokalen Minima in der Superposition der einzelnen LSFs, welche genau auf der halben diagonalen Verbindungsstrecke zwischen den LEDs liegen. Diese Punkte sind in Abbildung 4.55 markiert. Für die Distanz zwischen LEDs gilt also im Allgemeinen  $\forall r \in \{0, \dots, N_L^r - 1\} \wedge c \in \{0, \dots, N_L^c - 1\}$ :

$$\|\text{pos}(\text{LED}_{(r, c)}) - \text{pos}(\text{LED}_{(r, c \pm 1)})\|_2 < \|\text{pos}(\text{LED}_{(r, c)}) - \text{pos}(\text{LED}_{(r \pm 1, c \pm 1)})\|_2 \quad (4.22)$$

Somit wird mit der Annahme, dass die Leuchtdichte in der diagonalen Verbindung zweier LEDs stärker fluktuiert (höhere Maxima, kleinere Minima),  $U_2$  analog zu [SRX19] wie folgt definiert:

$$U_2 = \frac{L_{\Sigma, \text{all}}((\text{pos}(\text{LED}_{(r_{c-1}, c_{c-1})}) + \text{pos}(\text{LED}_{(r_c, c_c)})) / 2)}{L_{\Sigma, \text{all}}(\text{pos}(\text{LED}_{(r_c, c_c)}))} \quad (4.23)$$

Mit  $U_2$  wird also das Verhältnis zwischen der simulierten Leuchtdichte auf der halben diagonalen zweier LEDs im Zentrum berechnet und liefert ein gutes Maß für die Höhe Leuchtdichteschwankungen. Für die 6 Modelle sind in Abbildung 4.56 zur Verdeutlichung Querschnitte durch die in Abbildung 4.55 rot markierte Linie visualisiert. Die Schnittlinie verläuft diagonal durch die LED-Zentren. Gut erkennbar ist insbesondere  $M_1$  mit einer starken Abnahme der Leuchtdichte. Die dargestellten 9 LED-Zentren sind gut erkennbar. Auf der anderen Seite resultieren beispielsweise  $M_4 \dots M_6$  in einem homogenerem Backlight. Da die Modelle aufsteigend nach  $r_{50\%}$  sortiert sind, ist gut erkennbar, dass der Abfall zu den Rändern hin, für höhere  $r_{50\%}$

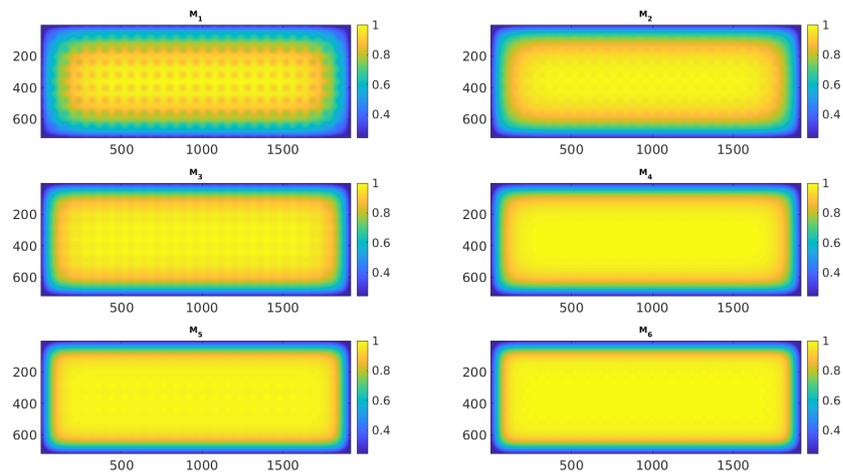


Abbildung 4.54.: Normierte Leuchtdichte simuliert aus obigen Modellen. Zur Visualisierung sind die Daten mit einer  $\gamma$ -Funktion gestreckt. Entnommen aus [SRX19].

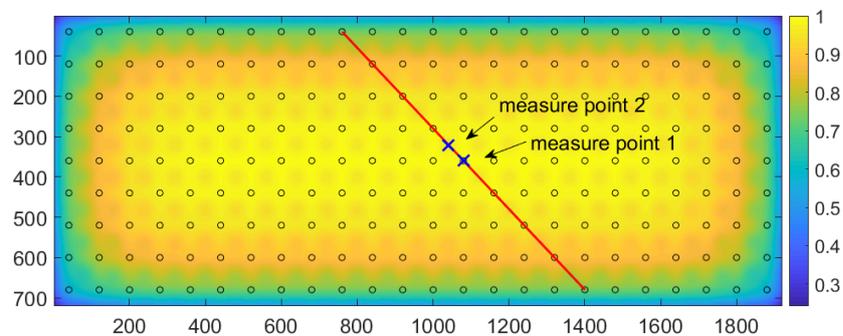


Abbildung 4.55.: Visualisierung der beiden Messpunkte zur Bestimmung von  $U_2$ . Entnommen aus [SRX19].

näher an den Rand rückt.

### Untersuchung der Robustheit

In Abschnitt 4.5.1 wurden bereits die Auswirkungen von ungenauer LED-Platzierung (LED-Jitter) während der BLU Fertigung auf die LSF untersucht. Analog dazu, wird in diesem Abschnitt die Auswirkung solch einer Fehlplatzierung auf die Uniformität der Modelle  $M_1 \dots M_6$  behandelt.

Auf die ursprünglichen LED-Positionen wird eine zufällige Abweichung von  $\pm 4\%$  des LED-Pitches gelegt und beispielhaft für  $M_2$  und  $M_4$  in Abbildung 4.57 visualisiert. In der Abbildung sind oben die resultierenden All-Ons basierend auf den selben LED-Positionen mit Jitter dargestellt. In dem unteren Teil sind Querschnitte durch die All-Ons entlang der roten Markierung gezeigt. Die gestrichelten blauen Linien zeigen dabei den ursprünglichen Zustand basierend auf korrekten LED-Positionen. Im All-On liefert  $M_4$  ein etwas homogeneres Ergebnis als für  $M_2$  und wird auch in den Querschnitten deutlich. Auch anhand lokaler Spitze-Spitze Abweichungen lässt sich die Aussage untermauern. Der Jitter wirkt sich also je nach LSF-Form unterschiedlich aus [SRX19].

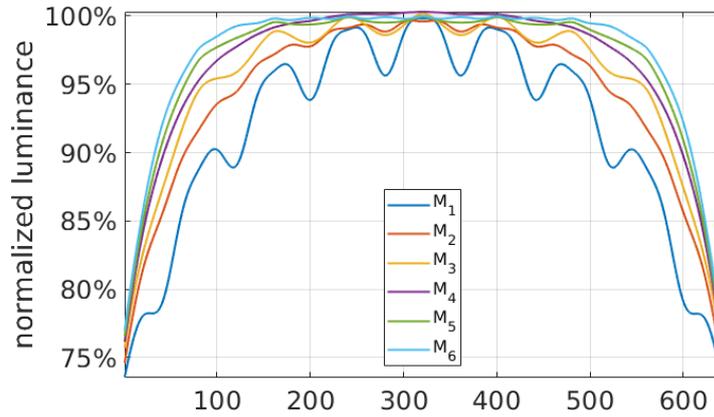


Abbildung 4.56.: Querschnitt durch die rote Linie aus Abbildung 4.55. Entnommen aus [SRX19].

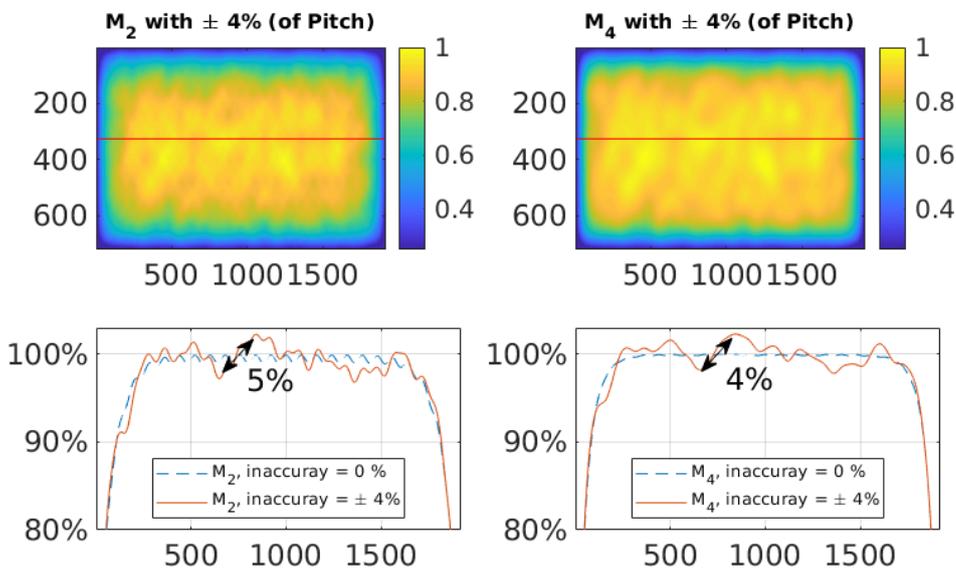


Abbildung 4.57.: Auswirkung auf die Uniformität bei nicht akkurater LED-Platzierung. Entnommen aus [SRX19].

Eine weitere Analyse basiert unter anderem auf der Kontrastempfindlichkeit der menschlichen Wahrnehmung. Hierfür eignet sich auch die Betrachtung von Steigungen. Zunächst wird jedoch, um später miteinander vergleichbare Ergebnisse zu erzielen, das All-On resultierend aus dem LED-Jitter mit dem als optimal gegeben All-On mit optimaler LED-Platzierung, normiert. Somit spielt das ursprüngliche All-On für die Betrachtung keine Rolle. Diese Normierung wird wie folgt definiert [SRX19]:

$$\tilde{L}_{\Sigma, \text{all}}(i, j) = \frac{L_{\Sigma, \text{all}}^{\pm 4\%}(i, j)}{L_{\Sigma, \text{all}}(i, j)} - 1, \quad \forall i \in \{0, \dots, N_P^r - 1\} \wedge j \in \{0, \dots, N_P^c - 1\} \quad (4.24)$$

Als Beispiel dient Abbildung 4.58 und zeigt die relative Abweichung als vertikalen Schnitt durch das Zentrum der BLU. Die höhere Abweichung von  $M_1$  und  $M_2$  im Vergleich zu den übrigen sticht aus diesem Plot hervor. Die Asymmetrie ist bedingt durch den zufälligen Jitter der LED-Positionen, welcher angewandt wurde.

Basierend auf Formel (4.24), wird eine mittlere Abweichung  $\bar{D}$  sowie der Mittelwert der absoluten Steigungen,

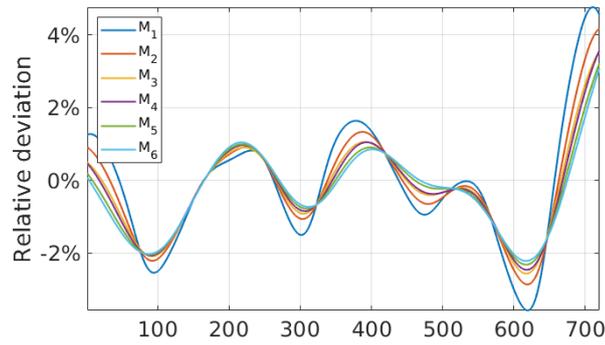


Abbildung 4.58.: Relative Abweichung als vertikaler Querschnitt durch die Mitte der BLU.

welche durch partielle Ableitungen repräsentiert werden ( $\bar{G}$ ), eingeführt. Die Berechnungsvorschriften sind in Formel (4.25) dargestellt [SRX19].

$$\bar{D} = \frac{\sum_{i=0}^{N_P^r-1} \sum_{j=0}^{N_P^s-1} \tilde{L}_{\Sigma,all}(i, j)}{N_P} \tag{4.25}$$

$$\bar{G} = \frac{\sum_{i=0}^{N_P^r-1} \sum_{j=0}^{N_P^s-1} \left( \left| \frac{\partial}{\partial i} \tilde{L}_{\Sigma,all}(i, j) \right| + \left| \frac{\partial}{\partial j} \tilde{L}_{\Sigma,all}(i, j) \right| \right)}{N_P}$$

Diese beiden Metriken können nun für alle Modelle berechnet werden. Das Resultat ist in Abbildung 4.59 dargestellt. Dabei ist auf der Abszisse das  $r_{50\%}$  Verhältnis aufgetragen. Die einzelnen Modelle sind markiert. Zu erkennen ist der fallende Trend von  $\bar{D}$  (rechte Ordinatenachse) und  $\bar{G}$  (linke Ordinatenachse) für höher werdende  $r_{50\%}$  und impliziert ein homogeneres Bild für höhere  $r_{50\%}$ .

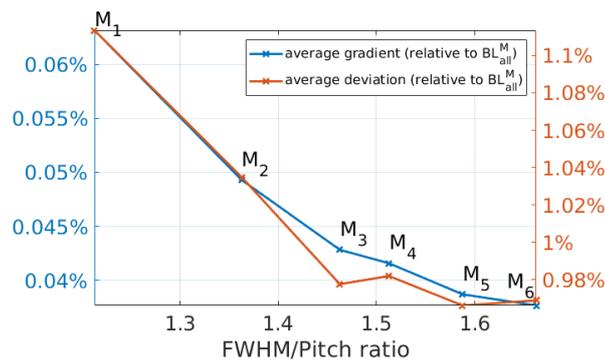


Abbildung 4.59.:  $r_{50\%}$  gegen  $\bar{D}$  (rot) und  $\bar{G}$  (blau).  $BL_{all}^M$  repräsentiert hier  $L_{\Sigma,all}$  eines Modells M. Entnommen aus [SRX19].

**Powersaving und LSF-Breite-zu-LED-Pitch Verhältnisse**

In diesem Abschnitt wurde die Robustheit einer LSF untersucht, wobei der Einfluss fest gewählt wurde. Dabei wurden die Auswirkungen der ungenauen Platzierung (*LED-Jitter*) deutlich. Zum weiteren Vergleich wurde die Reduktion des Leistungsbedarfs der verschiedenen Modelle anhand eines HMI-Testsets bestimmt (siehe Abschnitt 3.5.1). Die Ergebnisse sind in Tabelle 4.5 dargestellt. Beispielhaft sind in Abbildung 4.60 die resultierenden Backlights nach Formel (4.19) für das HMI-Testbild 3 (Anhang B) dargestellt.

Die Powersaving-Werte weichen hier für die verschiedenen Modelle nur geringfügig voneinander ab. Dieses Resultat erklärt sich dadurch, dass der maximale Einfluss für alle Modelle identisch gewählt wurde. Daraus kann geschlussfolgert werden (auch in Verbindung mit Abschnitt 4.5.1), dass für das Powersaving der maximale Einfluss entscheidend ist, während  $r_{50\%}$  für diesen Fall eine geringere Rolle spielt und für die Uniformität wichtiger ist. So wird insgesamt als Trade-Off ein maximaler Einfluss von  $\approx 30\%$  sowie ein  $r_{50\%} \geq 1,5$  vorgeschlagen, um einen effizienten Local-Dimming Algorithmus zu ermöglichen und eine adäquate Bildqualität zu liefern. Aus diesen konkreten Modellen  $M_1 \dots M_6$  wird  $M_4$  als Trade-Off empfohlen.

	$M_1$	$M_2$	$M_3$	$M_4$	$M_5$	$M_6$
Einsparung	25,8 %	26,4 %	26,7 %	26,6 %	26,9 %	26,9 %

Tabelle 4.5.: Reduktion des Leistungsumsatzes mit Local-Dimming.

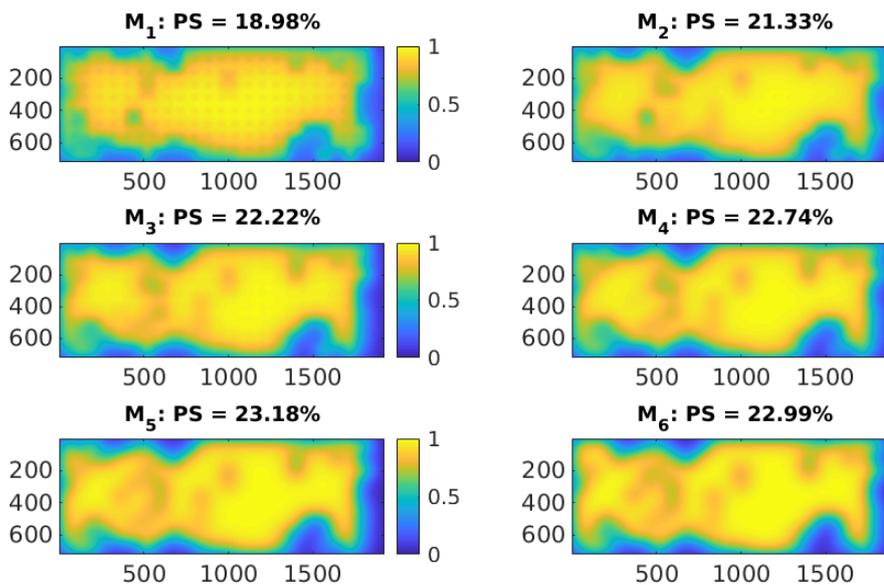


Abbildung 4.60.: Local-Dimming: resultierendes Backlight zum Testbild Nummer 3.

Zusätzlich zu dem  $r_{50\%}$  Verhältnis werden analog die Verhältnisse  $r_{90\%}$ ,  $r_{80\%}$  und  $r_{10\%}$  definiert und die Korrelation einem Uniformitätsmaß hergestellt. Da in diesem Abschnitt für die finale Selektion der Modelle bereits eine große Anzahl an Parametertripel simuliert (vgl. Abbildung 4.42) wurde, wird dieser Datensatz für die Analyse genutzt. Die Verhältnisse für die 6 ausgewählten Modelle sind in Tabelle 4.6 zusammengefasst. In Abbildung 4.61 a) und b) (Seite 125) sind auf der Abszisse die Verhältnisse der LSF Breite zu LED-Pitch aufgetragen. Die Ordinate zeigt hingegen  $1 - U_2$  nach Formel (4.23). Die Datenpunkte entsprechen denen

	$M_1$	$M_2$	$M_3$	$M_4$	$M_5$	$M_6$
$r_{10\%}$	2,70	2,92	3,10	3,10	3,15	3,15
$r_{50\%}$	1,21	1,36	1,46	1,51	1,59	1,66
$r_{80\%}$	0,75	0,82	0,82	0,87	0,90	0,95
$r_{90\%}$	0,55	0,57	0,57	0,60	0,62	0,65

Tabelle 4.6.: Verschiedene Breiten zu LED-Pitch Verhältnisse der 6 ausgewählten LSF-Formen mit festgelegtem maximalen Einfluss.

aus dem Datensatz. Hierzu wurde  $U_2$  ausgewertet und die Ergebnisse mit den jeweiligen Regressionsgeraden beschrieben. Leicht zu erkennen ist in a), dass  $r_{10\%}$  nicht zur Charakterisierung der Uniformität genutzt werden kann, da die Streuung zu hoch ist und somit keine Rückschlüsse auf die Uniformität gezogen werden können. Bei  $r_{50\%}$  ist die Streuung schon deutlich geringer. Die deutlich höheren Korrelationen ergeben sich jedoch für  $r_{80\%}$  bzw.  $r_{90\%}$ . In b) werden die beiden Verhältnisse nochmal dargestellt und zeigen einen deutlichen Trend an. Bei einem konstanten maximalen Einfluss führt ein höheres  $r_{80/90\%}$  zu einer höheren Uniformität  $U_2$ . Die diskret wirkenden Verhältnisse mit sichtbaren Abständen rühren von der Schrittweite der Parametervariation und wurden hier etwas gröber gewählt, um die Simulationszeiten zu verringern. Dieses Vorgehen ändert jedoch nicht den beobachteten Trend. In Tabelle 4.7 sind die resultierenden Pearson-Korrelationskoeffizienten zu den jeweiligen Verhältnissen dargestellt. Da  $r_{80\%}$  die höchste Korrelation bietet und die volle Weite bei 80 % einfacher zu bestimmen ist als  $r_{90\%}$  wird  $r_{80\%}$  als Maß für die Uniformität vorgeschlagen. Dies gilt es im BLU-Design Prozess zu berücksichtigen.

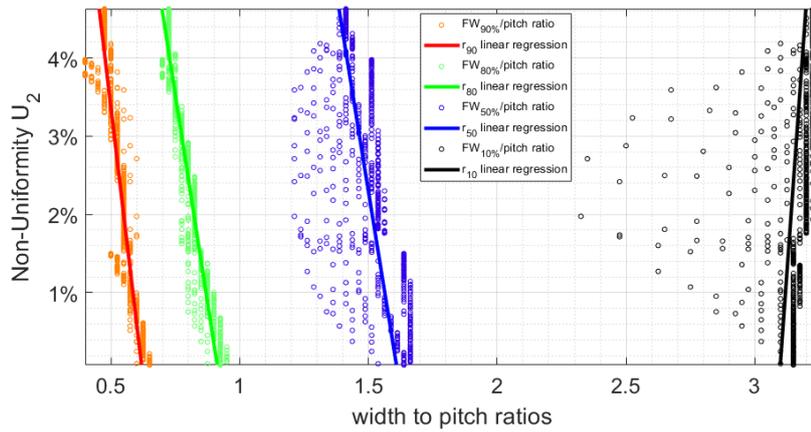
	$r_{10\%}$	$r_{50\%}$	$r_{80\%}$	$r_{90\%}$
Korrelationskoeffizient	0,235	-0,594	-0,942	-0,891

Tabelle 4.7.: Pearson-Korrelationskoeffizienten der verschiedenen LSF-Breiten zu Pitch Verhältnissen zu Uniformitätsmaß aus Abbildung 4.61.

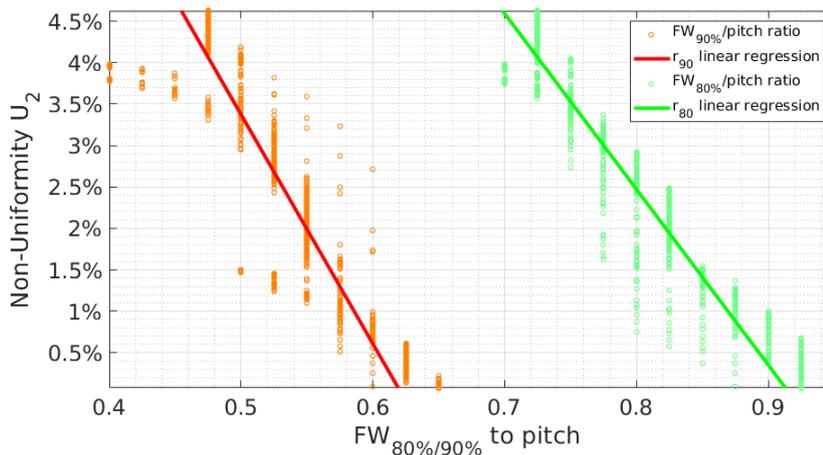
## 4.6. Zusammenfassung

In diesem Kapitel wurde die LSF-Form und deren Auswirkungen auf die Uniformität und Local-Dimming Performance untersucht. Ein deutlicher Unterschied zwischen Edge-Lit und Direct-Lit LSFs wurde anhand realer BLUs aufgezeigt und die jeweiligen Einflüsse miteinander verglichen. Des Weiteren wurde eine Definition der *Lokalität* eingeführt. Verschiedene LSF-Formen können somit mithilfe der definierten Lokalität auch miteinander verglichen werden. Die resultierenden Werte für die gezeigten, reale LSFs sind in Tabelle 4.8 dargestellt. Zu Erkennen ist der Trend zu höherer Lokalität in Abhängigkeit der steigenden LED Zahl. Erkennbar ist jedoch auch, dass BLUs mit der gleichen LED-Anzahl durchaus verschiedene Lokalitäten aufweisen können. So ist die BLU des Hisense TVs trotz höherer LED-Anzahl aufgrund eines *globaleren* Charakters, auf dem Niveau einer anderen BLU mit 133 LEDs.

In diesem Kapitel wurden verschiedene LSF-Formen modelliert, indem entweder der maximale Einfluss  $\check{L}^{\max}$  konstant gehalten und  $r_{50\%}$  variiert wurde und vice versa. Anhand dieser LSF-Modelle kann die Local-Dimming Performance und Uniformität des Backlights abgeschätzt werden. Basierend auf diesen



(a)  $r_{90\%}, r_{80\%}, r_{50\%}$  und  $r_{10\%}$  Verhältnisse vs. Nicht-Uniformität.



(b) Vergleich von  $r_{90\%}$  mit  $r_{80\%}$ .

Abbildung 4.61.: Diverse LSF-Breiten zu LED-Pitch Verhältnisse vs. Nicht-Uniformität  $1 - U_2$ .

	BLUs ( $N_L$ )					
	7 (Edge)	60 (aus 240)	133	240 (Hisense TV)	240 (Conti)	319
$\check{\Lambda}$	47,58 %	65,63 %	74,42 %	75 %	81,37 %	84,35 %

Tabelle 4.8.: Vergleich der aller bisheriger  $\check{\Lambda}$  angegeben in %.

Modellen wurde die *Robustheit* verschiedener Formen und deren Auswirkungen auf das Powersaving und Uniformität untersucht. Je größer  $r_{50\%}$ , desto uniformer jedoch auch globaler wird eine LSF. Es wird zudem der verbreitete Ablauf des BLU Designs ohne Berücksichtigung des Local-Dimming Systems eingeführt und festgestellt, dass heutige BLU-Designprozess im Allgemeinen nicht die optimale Local-Dimming Performance liefert, da Local-Dimming nicht „eindesigned“ ist, sondern nachrangig behandelt wird. Zur optimalen Verzahnung mit dem Local-Dimming System wird ein *erkenntnisorientierter* Ansatz des Design-Ablaufs vorgeschlagen.

Aufgrund der Ergebnisse aus den Unterkapiteln wird suggeriert, dass die LSF-Form beim BLU-Design Prozess berücksichtigt werden muss um einen Trade-Off zwischen Robustheit, Uniformität und Local-Dimming Performance herzustellen. Es wurde festgestellt, dass  $r_{80\%}$  zusätzlich zu der wichtigen Kennzahl  $r_{50\%}$ ,

aufgrund einer Korrelation eine wichtige Rolle bei der Auswirkung auf die Uniformität spielt. Im Allgemeinen wird ein  $r_{50\%} \geq 1,5$  und ein maximaler Einfluss von  $\approx 30\%$  empfohlen [Sch+19b; SX19; SRX19].

### Powersaving und LED-Anzahl

Wird nun anhand eines LSF-Modells das  $r_{50\%}$  Verhältnis auf 1,5 gesetzt und die LED Anzahl erhöht, ergibt sich für ein gemischtes automotive Testset und natürliche Bilder, die Reduktion des Leistungsbedarfs dargestellt in Abbildung 4.62. Auf der Abszisse ist hier  $\sqrt{n_{LEDs}}$  aufgetragen, um die Zweidimensionalität der LED-Matrix Rechnung zu tragen. Als Beispiel wie das resultierende Backlight aussehen kann, sind in Abbildung 4.63 zu 3 ausgewählten Testbildern jeweils absteigend, das Backlight mit 96, 319 und 2400 LEDs dargestellt. Die steigende LED-Anzahl führt also zu einem besseren Abbild der Bilddaten im Backlight,

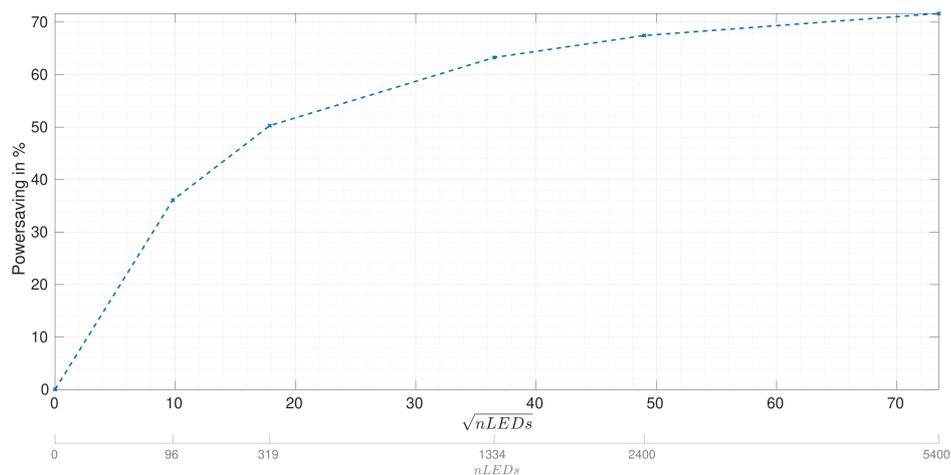


Abbildung 4.62.: Resultierendes Powersaving von der Quadratwurzel der LED-Anzahl. Entnommen aus [Sch+19b].

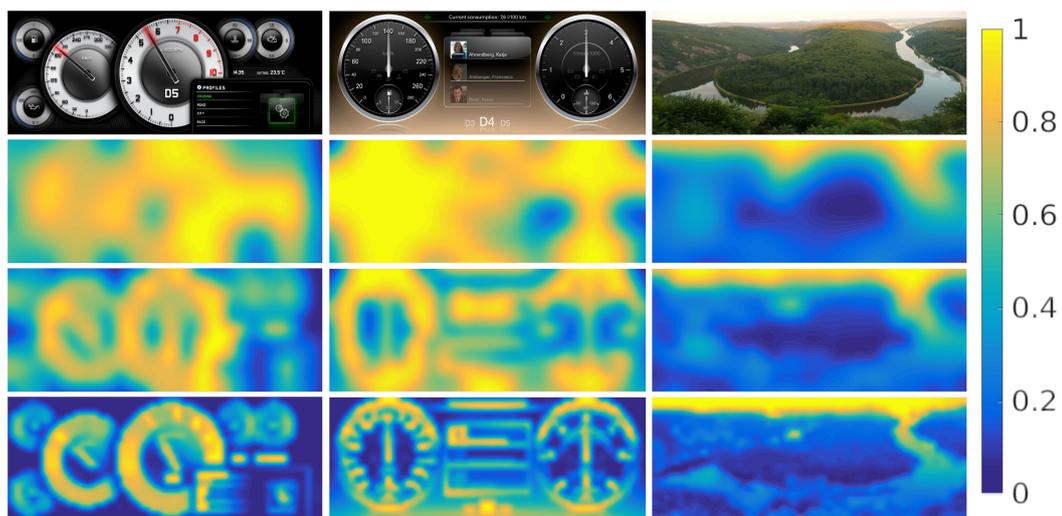


Abbildung 4.63.: Beispiele simulierter Backlights mit unterschiedlicher LED-Anzahl. Jeweils 96, 319 und 2400 LEDs (von oben nach unten). Teilweise entnommen aus [Sch+19b].

welches dadurch genauer wird. Diese Verbesserung wird auch anhand des Anteils der schwarzen Pixel nach Formel (4.11) (Seite 98) deutlich. Für diese Auswertung werden alle Pixel gezählt, deren Leuchtdichte  $< 1/K \cdot L_{\max}$  ist.  $K$  stellt hier das Kontrastverhältnis  $K : 1$  und  $L_{\max}$  die maximale Leuchtdichte dar. Das Ergebnis ist in Tabelle 4.9 für ein Bild gezeigt und mit den schwarzen Pixeln eines OLED Displays verglichen [Sch+19b]. Der Trend führt wie zu erwarten bei steigender LED-Anzahl zu einem deutlichen Anstieg der schwarzen Pixel. Der Unterschied zwischen 5400 LEDs und OLED ist aus der physiologischen Sicht, nicht mehr so groß, da auch im Auge durch Streuung [Bau13] helle Objekte auf einem OLED Display schwarze Flächen um das Objekt mit einem Schleier belegen. Aufgrund der langen Lebenszeit der LCDs sowie der höheren Leuchtdichte im Vergleich zu OLED Displays, hat ein LCD immer noch signifikante Vorteile zur Anwendung im Automobil.

$N_L$	96	319	1334	2400	5400	<b>OLED</b>
$P_{\text{black}}$	0,00 %	1,02 %	6,36 %	9,54 %	14,15 %	29,65 %

Tabelle 4.9.: Anzahl schwarzer Pixel  $< 1/K \cdot L_{\max}$  des HMI Testbildes 3 bei verschiedener LED Anzahl.

Die steigende BLU Auflösung hat nicht nur wie hier gezeigt Auswirkungen auf die LSF-Form oder Uniformität, sondern hat auch direkten Einfluss auf die Hardwarekomplexität des Local-Dimming Systems. Es werden neue Methoden benötigt um die daraus entstehenden Limitierungen und Herausforderungen zu meistern. Diese werden im nächsten Kapitel vorgestellt und neue Lösungen vorgeschlagen.



# **5. Beschleunigte Optimierung, LSF-Modellkonzept und Systemarchitektur für eine effiziente Hardware-Implementierung des Local-Dimming Algorithmus**

Dieses Kapitel widmet sich der Entwicklung neuartiger Ansätze zur Realisierung eines Local-Dimming Systems, um die LCD Qualität speziell für automobiler Anwendung sowie Reduktion der Leistungsaufnahme zu erhöhen und die Kosten zu senken.

Die bisherigen SoA Verfahren stoßen an die Grenzen der Wirtschaftlichkeit in Bezug auf die Hardwarekosten und Umsetzbarkeit. Die Kosten des Local-Dimming Prozessors werden durch die gestiegenen LED-Zahlen in der BLU in die Höhe getrieben. Des Weiteren sind einige visuelle Probleme zu lösen, die im Zusammenhang mit Local-Dimming automotiver Anwendungen stehen. Sie werden in Kapitel 6 gesondert behandelt.

Der bereits eingeführte SSC Algorithmus wurde schon in vielen Edge-Lit Anwendungen in Pixelpipeline basierenden Hardware Implementierungen erprobt und wird hier als SoA (vgl. Abschnitt 3.5) betrachtet. Insbesondere für die Berechnung der LED-Werte bedarf es folglich neuer Ansätze welche in den kommenden Abschnitten vorgestellt werden.

## **5.1. Auswirkungen und Herausforderungen hoher LED-Auflösungen auf den Local-Dimming Algorithmus**

In diesem Abschnitt wird auf die Auswirkungen der steigenden LED-Anzahl in der BLU (vgl. Kapitel 4) und der daraus resultierenden Herausforderungen und notwendigen Optimierungen des aktuellen Standes der Technik des SSC eingegangen.

### **Allgemeine Herausforderungen**

Wie in Abschnitt 3.1 kurz eingeführt, kommen im Fahrzeug typischerweise künstliche HMI-Inhalte mit hohem Kontrast und harten Übergängen vor. Die Farbörter müssen für die Qualitätserhaltung gut getroffen werden. Folglich muss die Pixelkompensation sehr genau sein, um Artefakte zu vermeiden. Zudem sollte die technische Umsetzung eine gewisse Flexibilität aufweisen (vgl. Abschnitt 5.5), um verschiedene BLU/Treiber Konstellationen abdecken zu können. Eine weitere Herausforderung stellt die oft hohe Lokalität der LSFs bei steigender LED-Anzahl dar. In Kombination mit typischen HMI Inhalten wird eine höhere Condenser Auflösung sowie eine größere Anzahl an Sample-Points bei der Backlicht-Berechnung

benötigt. Zudem kommen zeitliche Limits hinzu. Darauf wird in den Abschnitten 5.1.1 und 5.1.2 eingegangen.

Wenn nicht anderes angegeben, beziehen sich im Folgenden die Angaben auf ein Panel mit dem Seitenverhältnis 8:3, welches sehr oft für das Instrument Cluster Display (ICD, Kombi-Instrument Anzeige) verwendet wird und die später eingeführten ICD Prototypen auf diesem Seitenverhältnis basieren.

### 5.1.1. Condenser- und Backlight Sample-Point Auflösung

In Abschnitt 3.5.1 wurde bereits die Funktion des *Condensers* eingeführt. Da die Bildkondensierung das Eingangsbild und somit die Zielwerte repräsentiert, auf welches das Backlight optimiert werden soll, ist diese zum größten Teil für die resultierende Reduktion des Leistungsbedarfs verantwortlich. In Abbildung 5.1 sind zu einem HMI-Testbild mehrere verschiedene Condenserauflösungen im Vergleich dargestellt. Oben links sind

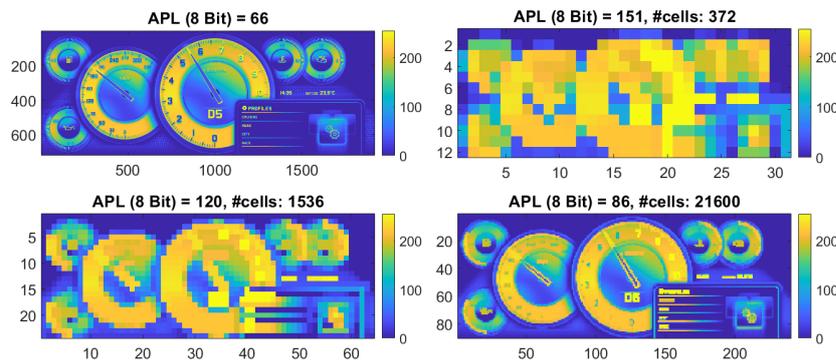


Abbildung 5.1.: Verschiedene Condensergrößen im Vergleich.

die maximalen Subpixel des Testbildes abgebildet. Es ist zu erkennen, dass der Average Pixel Level (APL) für kleiner werdende Condenserezellen – also bei steigender Condenserezellen-Anzahl – sich dem APL des original Bildes nähert. Auch in [SWF12] wird festgestellt, dass ab einer gewissen Anzahl der Condenserezellen, die Abweichung für Local-Dimming vernachlässigbar wird. Da die Pixel innerhalb einer Condenserezelle durch einen einzigen Wert repräsentiert werden ist ersichtlich, dass einzelne hohe Werte dazu führen können, dass die gesamte Zelle deutlich über dem APL der zugehörigen Pixel liegt. Dies kann schließlich zu einer geringeren Reduktion des Leistungsbedarfs führen. In Abbildung 5.2 ist die Abhängigkeit des APL von der Quadratwurzel der Condenserezellen ( $\sqrt{N_C}$ ) für das obige Testbild dargestellt. Die Knicke, insbesondere bei

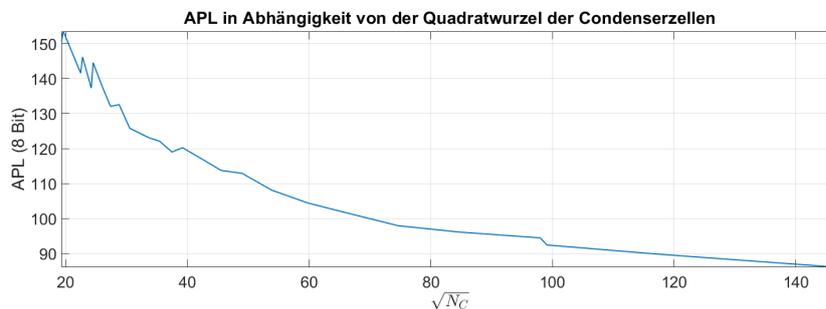


Abbildung 5.2.: APL in Abhängigkeit von der Quadratwurzel der Condenserezellenanzahl. Original APL = 66.

großen Condensergrößen (inbegriffene Pixel) führt die grobe Condenserauflösung dazu, dass Pixel aus einer hellen Zelle durch die neuen Grenzen gerade so noch in eine andere Zelle wandern, welche generell dunkler sind.

In Abschnitt 4.1.1 wurde gezeigt, wie unterschiedlich verschiedene LSFs und die resultierenden Einflüsse sein können. Insbesondere bei Edge-Lit BLUs reicht eine kleine Zellenzahl aus, da der Einfluss sich nur langsam (Steigung) ändert und folglich keine starken Änderungen innerhalb der Zellen vorkommen. Geläufig sind für diesen Fall 558 Zellen bei einem Seitenverhältnis von 16:9. Bei 8:3 entspricht die Anzahl 372 Zellen (Beispiel im Bild oben). Bei Direct-Lit BLUs können jedoch diese Änderungen innerhalb einer zu groß ausgelegten Zelle vorkommen und damit nicht nur das Powersaving, sondern auch die visuelle Qualität beeinflussen. Wird beispielsweise ein zu hoher Einfluss angenommen, um sicherzustellen, dass die ursprüngliche Leuchtdichte erreicht wird, so liegen die resultierenden LED Werte nicht im Optimum, sondern werden deutlich höher berechnet, was zu einem suboptimalen Powersaving führt und damit die Vorteile eines Direct-Lit Gerätes nicht ausgeschöpft werden können.

Zusätzlich sind beim SoA wie in [Alb10; Jun14; Sch17] beschrieben, die einzelnen Einflüsse der LEDs basierend auf Stützstellen (*Sample-Point*) gespeichert und liegen in derselben Auflösung vor, wie auch die Condenserezellen. So kann das Backlight direkt *on-the-fly* während der Berechnung der LED-Werte in den Sample-Point RAM gespeichert werden. Im Postprocessing werden diese Sample-Points genutzt und für jedes Pixel *bilinear* interpoliert, um die Pixelkompensation durchzuführen. In Abbildung 5.3 ist die zentrale LSF einer 319 BLU dargestellt. Die 372 Sample-Points sind in diesem Fall zu wenig, da das

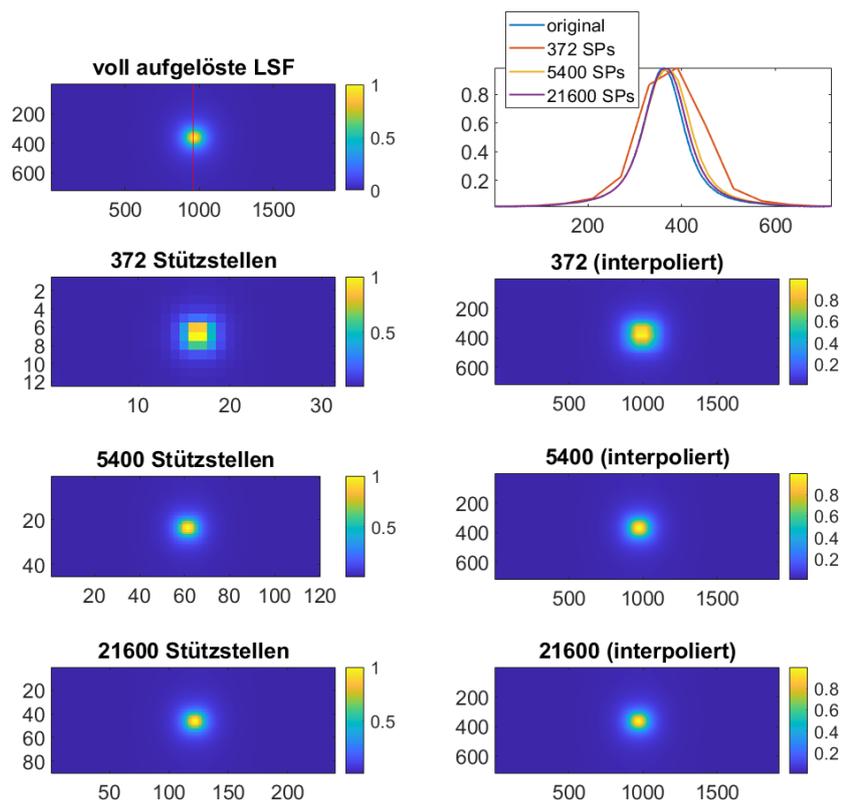


Abbildung 5.3.: Beispiel einer Sample-Point basierenden LSF Speicherung einer Direct-Lit LSF sowie die resultierende bilineare Interpolation.

interpolierte Bild eine eher rechteckige Form aufweist und zu weit ausgedehnt ist. Dadurch kann es bei der Pixelkompensation zur Kantenbildung kommen, da Kompensationsfaktor aus den Interpolationen gewonnen werden. Zu niedrig aufgelöste Stützstellen führen auch dazu, dass zu hohe Einflüsse angenommen werden müssen, um die erforderliche Leuchtdichte zu garantieren. Dadurch werden die LEDs zu hell und die Reduktion des Strombedarfs nicht maximal.

Für einen weiteren Überblick über die resultierende Abweichung des interpolierten Backlights (hier einer LSF) von der voll aufgelösten LSF, ist in Abbildung 5.4 der Root-Mean-Square Error (RMSE, Wurzel der mittleren Fehlerquadratsumme) in Abhängigkeit von der Quadratwurzel der Conderzellen-Anzahl ( $\sqrt{N_C}$ ) aufgetragen und mit einem polynomiellen Fit 2. Grades zum Verdeutlichen des Trends angepasst. Im Vergleich stehen die 319er Direct-Lit BLU und die vorgestellte Edge-Lit BLU. Um eine Vergleichbarkeit zu gewährleisten werden die Daten zunächst jeweils auf ihren Maximalwert normiert. Der RMSE wird dann basierend auf den normierten Werten berechnet. Um die weniger signifikanten Abweichung nicht in die Berechnung aufzunehmen, werden nur die Positionen im Bild berücksichtigt, welche von diesen LEDs ihren maximalen Einfluss erfahren (Positionen  $p$  mit Haupt-LED  $l$ ,  $LED_{p,l}^M$ ). Zum Visualisieren der Zuordnung wird auf Abbildung 4.9 (Seite 82) verwiesen. Die erkennbare Fluktuation für kleine  $\sqrt{N_C}$  ist hier analog zu der in

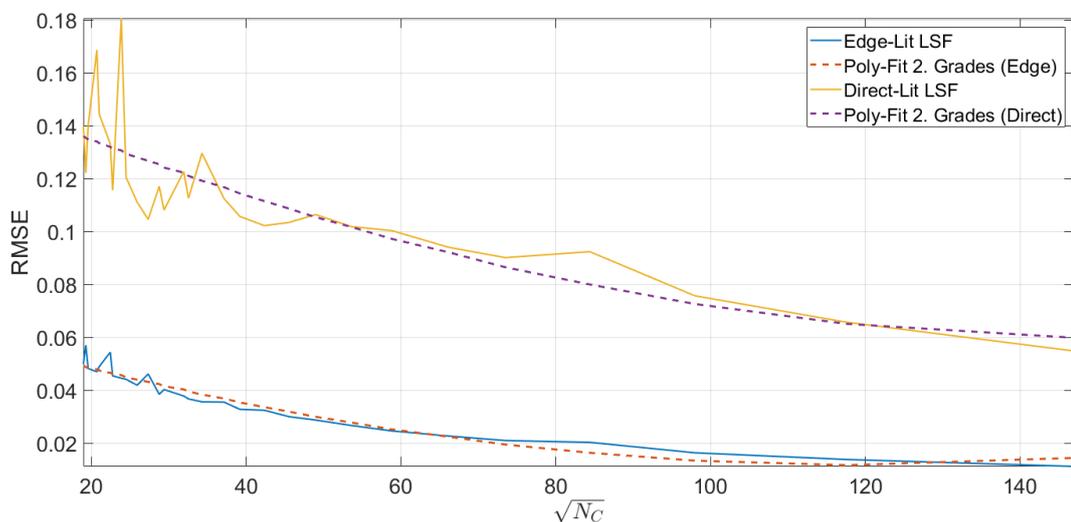


Abbildung 5.4.: RMSE resultierend aus Sample-Point basierendem LSF Speicherung mit anschließender bilinearen Interpolation.

Abbildung 5.2 zu erklären.

Eine wichtige Erkenntnis aus dieser Grafik ist, dass Direct-Lit LSFs eine höhere Sample-Point Anzahl benötigen um den resultierenden Fehler bezogen auf die Positionen mit dem Haupteinfluss, zu reduzieren. Der Fehler hängt sehr stark von den Ableitungen der LSFs ab. Würde der Fehler auf der gesamten Fläche berücksichtigt, so wird er deutlich geringer, da viel Fläche nur einen sehr geringen Lichtbeitrag erfährt und schließlich im Rauschen verschwindet (Genauigkeit des genutzten Datentyps).

Da wie oben erläutert, die Sample-Point Anzahl identisch zu der Condensierzellen-Anzahl ist, müssen diese für eine LSF mit höherer Lokalität weiter erhöht werden, was sich folglich auf den Hardwarebedarf und benötigte Rechenzeit auswirkt (siehe Abschnitt 5.1.2). An dieser Stelle muss ein Trade-Off eingegangen werden.

### 5.1.2. Grenzen der Hardware-Implementierung und resultierende Herausforderungen

Neben dem LED bedingten Anstieg, beeinflusst die Anzahl der Condenser-Zellen ebenfalls die Größe der Pixelorder. Die SoA Pixelorder des SSC wurde bereit in Abschnitt 3.5.1.2 vorgestellt. Die Struktur der Pixelorder wird für die folgende Berechnung wie eingeführt angenommen. Im vorangegangenen Abschnitt wurden bereits die Gründe genannt, die dazu führen, dass die Condensierzellen-Anzahl für steigende LED-Zahlen erhöht werden muss. Dies führt schließlich dazu, dass die Pixelorder mehr Einträge bekommt (Anzahl der Condensierzellen). Da das Verfahren zur Berechnung der LED-Werte auf dieser Pixelorder basiert und die Einträge nacheinander abgearbeitet werden, erhöht sich die Anzahl der benötigten Clock Zyklen drastisch. In Abbildung 5.5 sind die resultierenden Berechnungszeiten gegen die Quadratwurzel der LED-Anzahl aufgetragen. Für die Betrachtung einer Condensierzelle werden 60 Takte benötigt. Beispielhaft werden 4 verschiedene Condensierzellen-Anzahlen gezeigt. Zusätzlich ist zum Verdeutlichen des zeitlichen Rahmens,

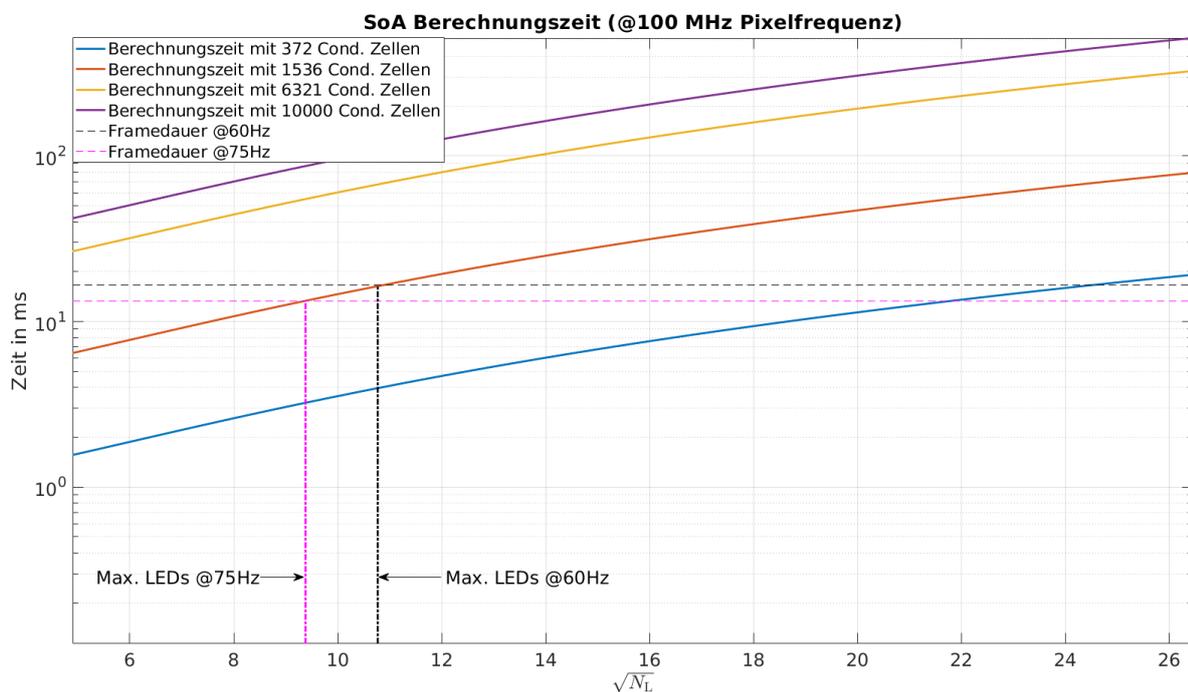


Abbildung 5.5.: SSC Berechnungszeit vs. Quadratwurzel der LED-Anzahl für verschiedene Condensergrößen.

die Zeiten für eine Bild mit der Bildwiederholfrequenz von 60 Hz bzw. 75 Hz für das Beispiel mit 1536 Condensierzellen markiert. Die harte zeitliche Grenze stellt immer die Zeit pro Bild für eine gegebene Bildwiederholrate dar. Es ist zu erkennen, dass der SSC nach SoA für eine höhere Condensierzellen-Anzahl, die Berechnungen nicht innerhalb eines Bildes fertigstellen kann.

Neben der zeitlichen Komponente kommt hinzu, dass der Speicherbedarf (in FPGAs und ASICs als SRAM realisiert) für die Pixelorder stark ansteigt. In Abschnitt 3.5.1.2 wurde bereits die Struktur der Pixelorder vorgestellt und erläutert wie Einflussmatrix in dieser Struktur gespeichert wird. Die steigende LED-Anzahl hat hierbei Auswirkungen auf die Wortbreite und die Condensierzellen-Anzahl auf die Adressen. In Abbildung 5.6 wird dieser steigende Bedarf in Abhängigkeit der Quadratwurzel der LED-Anzahl für

verschiedene Condensergrößen gezeigt. Insbesondere für die Sample-Point gestützte LSF muss die Zellgröße deutlich steigen und führt schnell dazu, dass der Speicherbedarf höher wird, als der auf dem FPGA vorhandenen Block-RAM. Beispielsweise verfügt das größere Xilinx Kintex 7 XC7K325T FPGA 16 Mbit an Block RAM [Xil18], während das größte automotiv zertifizierte Xilinx Artix XA7A100T nur 4,8 Mbit besitzt [Xil17]. Es sei noch erwähnt, dass in dieser Betrachtung der Speicher für Gamma, Degamma und andere Funktionen Kurven nicht berücksichtigt wurden. Die leichten auffälligen Knicke in der Abbildung kommen an den Grenzen der Zweierpotenzen vor. An den Knicken wird jeweils ein Bit mehr benötigt, um die LED Nummer in der Pixelorder zu speichern. Die benötigten Bits berechnen sich zu

$$B(N_L) = \lceil \log_2 N_L \rceil \quad (5.1)$$

Zusätzlich zum Speicherbedarf der Pixelorder steigt mit steigender Condenserezellen-Anzahl ebenfalls der

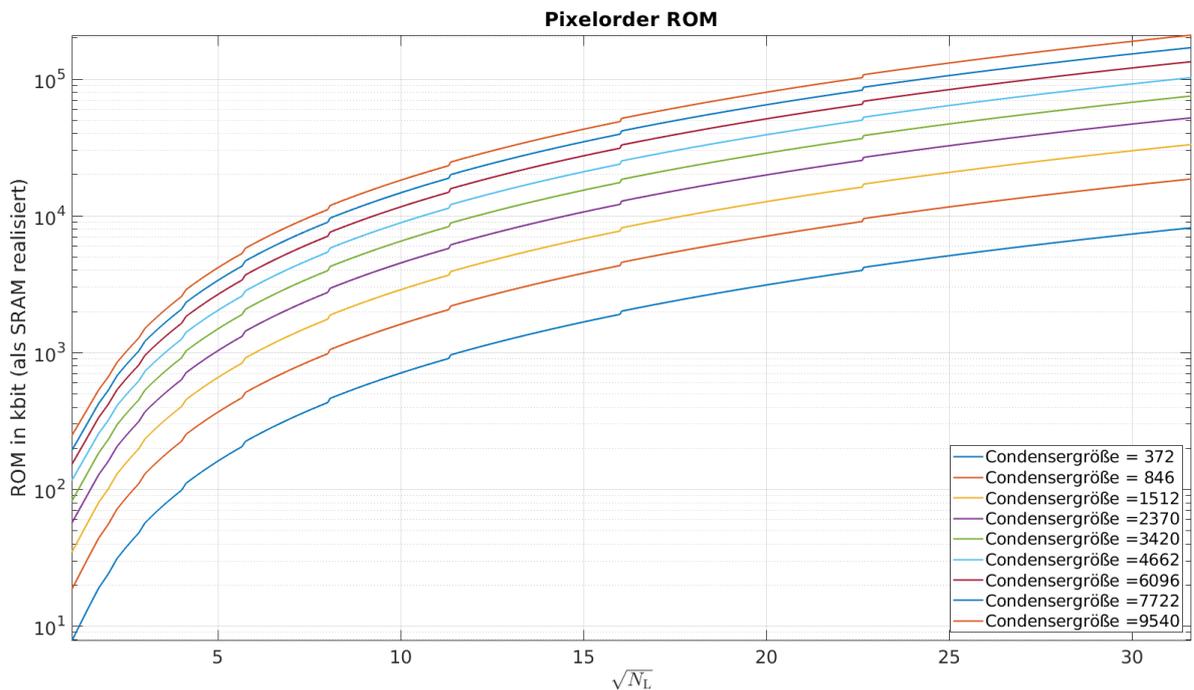


Abbildung 5.6.: Pixelorder ROM (als SRAM) Bedarf vs. Quadratwurzel der LED-Anzahl für verschiedene Condensergrößen.

benötigte SRAM für Condenser und die Sample-Points um die gestiegenen BLU-Auflösung auch zur weiteren Qualitätssteigerung nutzen zu können und führt somit unweigerlich zu einem sehr hohen Anstieg der Speicherkosten. Unter Annahme einer 10 Bit Auflösung wird der folgende SRAM aus Tabelle 5.1 benötigt. Dieser SRAM ist unabhängig von der Anzahl der LEDs. Berücksichtigt wurde hier ebenfalls der doppelte SRAM, da die SoA Implementierung den RAM abwechselnd (pro Frame) zum Speichern bzw. Berechnen und Kompensieren aufgrund zeitlicher Einschränkung nutzt.

Aufgrund des schnell steigenden SRAM Bedarfs und der zeitlichen Limitierung durch die FPS, kann der Local-Dimming Algorithmus wie er für Edge-Lit BLUs eingesetzt wird, nicht so adaptiert werden. Es werden neue Konzepte und Architekturen notwendig.

In den Abschnitten 5.2 bis 6.5 werden dazu neue Methoden vorgestellt und entwickelt um die resultierenden

	Condensierzellen-Anzahl								
	372	846	1512	2370	3420	4662	6096	7722	9540
SRAM in kbit	14,88	33,84	60,48	94,80	136,80	186,48	243,84	308,88	381,60

Tabelle 5.1.: SRAM Bedarf der SoA Implementierung für den Condenser- und SP-RAM in kbit.

Herausforderungen zu bewältigen. Es wird auch auf die neue abgeleitete Local-Dimming Systemarchitektur (Hardwaredesign) eingegangen und für die Prototypen realisiert.

## 5.2. Konzept für eine effiziente Optimierung

Wie in den vorangegangenen Abschnitten dargestellt wurde, sind die Qualität als auch Hardwarekosten für die automobiler Anwendung entscheidend. Mit Hinblick auf die in Abschnitt 5.1 eingeführten Auswirkungen von hoch-auflösenden BLUs auf die zeitlichen Limits und Speicherbedarf ist die Notwendigkeit alternative Methoden zur LED-Berechnung zu entwickeln gegeben. Dabei ist das Ziel eine Pixelpipeline basierende Hardwarearchitektur zu ermöglichen und aus Sicherheits- und Kostengründen auf Framebuffer zu verzichten. Die Abwesenheit von Framebuffer ermöglicht es, den Local-Dimming Prozessor mitten in eine vorhandene Pixelpipeline zu positionieren und stellt auch einen wichtigen Unique Selling Point (USP, Alleinstellungsmerkmal) dar.

In Abschnitt 4.1.1 wurden die Unterschiede von Direct-Lit zu Edge-Lit LSFs aufgezeigt. Diese Unterschiede in Form und Anzahl haben eine komplexe Auswirkung auf Local-Dimming Algorithmen. Das bisherige Optimierungsverfahren reicht folglich nicht aus. Vielmehr müssen mehrere neuartige Methoden und das Zusammenspiel des gesamten Systems konzipiert werden. Beim SoA ist die Berechnungszeit und LSF-Speicher alleinig von der *Pixelorder* abhängig, welche ausschließlich mit der Pixelfrequenz (*Pixelclock*) arbeitet (siehe Abschnitt 3.5.1.2). Die Anzahl der Condensierzellen ( $N_C$ ) entspricht der Anzahl der Sample-Points ( $N_{SP}$ ) für die Backlight Berechnung. In Abbildung 5.5 auf Seite 133 wird ersichtlich, dass beispielsweise für 1536 Condensierzellen das zeitliche Limit schon bei unter 121 LEDs erreicht ist. 1536 Zellen stellen darüber hinaus noch für die meisten LSFs eine zu geringe Anzahl für eine adäquate Pixelkompensation dar, da manche LSFs eine sehr steile Steigung vorweisen. Folglich müsste diese Anzahl weiter erhöht werden und führt wiederum unweigerlich zur Erhöhung der Berechnungszeit und befindet sich somit schon bei  $N_C = N_{SP} = 6321$  außerhalb der zeitlichen Limits und bei einem SRAM Bedarf, welcher die meisten FPGAs nicht im IC vorweisen können. Diese Einschränkungen begründen die in den nächsten Abschnitten eingeführten Optimierungen und Methoden zur Reduktion der Berechnungszeit und Speicherbedarfs.

Die wesentlichen erforderlichen Züge der notwendigen Veränderungen stellen also

- Lösen des zeitlichen Problems
- die Auflösung der Bindung  $N_C = N_{SP}$
- Lösen der Speichereinschränkung

dar, welche miteinander stark gekoppelt sind. Folglich wird ein neuartiger Ansatz zur kostengünstigen und effizienten Realisierung in Hardware benötigt, ohne auf die Berücksichtigung des *Crosstalks* zu verzichten.

### 5.2.1. Methode zur beschleunigten Optimierung

Ein Ansatz zur Lösung des zeitlichen Problems wird hier vorgestellt. Das Konzept ist die Begrenzung der LED-Anzahl, welche im Rahmen der LED-Berechnung berücksichtigt wird. Dieses Vorgehen wird durch die LSF-Formen wie in Abschnitt 4.1.2 vorgestellt wurden, begünstigt. Eine für Local-Dimming geeignete LSF sollte ab einem bestimmten Abstand ( $d_{th}$ ) vom LED-Zentrum nur noch einen sehr geringen Beitrag zur Leuchtdichte hinter einem Pixel beisteuern. Dadurch verringert sich die Anzahl der LEDs mit signifikantem Lichtbeitrag. Das Prinzip des effizienten Modellierens des *Crosstalks* wurde in [Sch+17] vorgestellt. Dieser Ansatz führt zur Reduktion der Berechnungszeit.

Somit gilt für die Abstände ( $d_{p,l}$ ) des Pixels  $p$  von der LED  $l$ :

$$\forall p \in \mathcal{P}, l \in \mathcal{L} : d_{p,l} > d_{th} \rightarrow \check{L}_{p,l} \approx 0 \quad (5.2)$$

Infolgedessen wird das ursprüngliche Optimierungsproblem wie in Abschnitt 2.5.3 beschrieben wurde, zu der Formulierung in Formel (5.3) angepasst um die reduzierte Anzahl der betrachteten LEDs zu berücksichtigen. Dabei sei  $\mathcal{M}_{i,j}$  die Menge der berücksichtigten LEDs im Pixel beziehungsweise Condensierzelle  $(i, j)$  und  $|\mathcal{M}_{i,j}| = \tilde{N}_L$ . Der Vektor  $\mathbf{v}$  enthält die einzelnen LED-Werte und  $\mathbf{c}$  repräsentiert die Pixel- bzw. Condenserwerte.

$$\min \left\{ \sum_{l=0}^{N_L-1} \mathbf{v}(l) : \sum_{m \in \mathcal{M}_{i,j}} \check{L}_{(i,j),m} \cdot \mathbf{v}(m) \geq v_{(i,j)} \cdot \mathbf{c}(i, j) \right\} \quad (5.3)$$

Da für die bei der Begrenzung der berücksichtigten LEDs gilt:  $\tilde{N}_L \ll N_L$  wird ersichtlich, dass die Condenserwerte  $\mathbf{c}(i, j)$  nicht für jeden Fall erreicht werden können. Dieser Umstand wird durch einen positionsabhängigen Faktor,  $v_{(i,j)} \in [0; 1]$  einbezogen und entspricht rechnerisch

$$v_{(i,j)} = \sum_{m \in \mathcal{M}_{i,j}} \check{L}_{(i,j),m}$$

Es gilt außerdem

$$\begin{aligned} \lim_{\tilde{N}_L \rightarrow N_L} v_{(i,j)} &= 1 \\ \text{und} & \\ \lim_{d_{th} \rightarrow 0} v_{(i,j)} &= 1 \end{aligned} \quad (5.4)$$

und führt zu dem idealerweise erreichbaren Wert  $\mathbf{c}(i, j)$ . Die LSF-Form spielt hier also eine sehr wichtige Rolle und entscheidet unter anderem über ein geeignetes  $\tilde{N}_L$ . Unabhängig der LSF-Form, bestimmen die zeitlichen Beschränkungen (auch die Clockfrequenz  $f_p$ ), die maximale LED-Anzahl ( $N_L$ ), die Anzahl der Condensierzellen ( $N_C$ ) und auch die Iterationszahl ( $N_{iter}$ ). Diese gibt an, wie oft über die Pixelorder in Hardware iteriert wird. Beispielsweise wird bei  $N_L = 1000$  und einer Berücksichtigung von  $7 \times 7$  LED-Nachbarn, die Berechnung auf 5% der ursprünglichen Komplexität reduziert [Sch+17]. Das Konzept wird in Abbildung 5.7 als Momentaufnahme während der Prozessierung dargestellt. Die roten Kreuze entsprechen hierbei den LED-Position und das rote Quadrat der aktuell betrachteten Condensierzelle. Die grünen Markierungen zeigen für diesen Fall die LEDs aus  $\mathcal{M}_{i,j}$  mit  $\tilde{N}_L = 25$ . Somit ist ein wichtiges neuartiges Merkmal des Local-Dimming Algorithmus die reduzierte Betrachtung der um eine Condensierzelle umliegende LED-Nachbarschaft. Dadurch

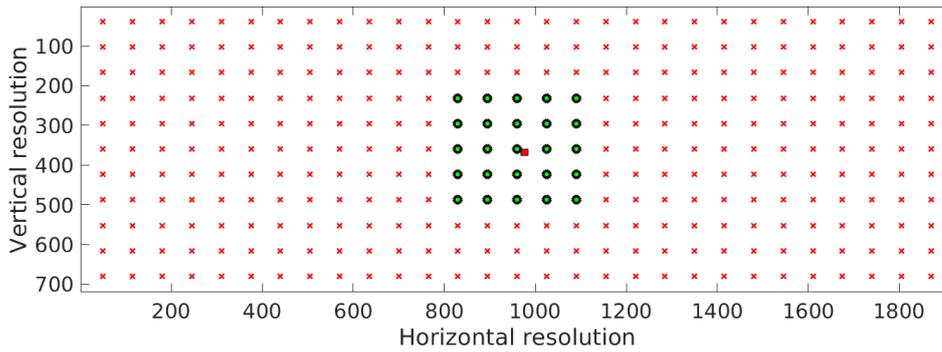


Abbildung 5.7.: Markierte LEDs aus  $\mathcal{M}_{i,j}$  mit  $|\mathcal{M}_{i,j}| = 25$  für die rote Condensierzelle.

wird die Berechnungszeit trotz der hohen LED-Anzahl begrenzt, da die Wortbreite der Pixelorder limitiert wird und ermöglicht damit die zeitlichen Constraints (verursacht durch hohe  $N_L$ ) einzuhalten.

### 5.2.2. Betrachtung der Auswirkungen einer reduzierten LED-Auswahl

Im vorherigen Abschnitt wurde eine wichtige Methode vorgestellt, um die LED-Berechnungszeit (*Optimierung*) zu reduzieren. Dabei muss jedoch ein Trade-Off eingegangen werden, da je nach LSF, die LED-Werte durch nicht berücksichtigte Licht-Beiträge von weiter entfernter liegenden LEDs tendenziell heller berechnet werden müssen, um die Zielleuchtdichte zu erhalten. In diesen Fällen fehlt das *Wissen* über die bisher berechneten LED-Werte der nicht betrachteten LEDs zugunsten der Berechnungszeit.

In diesem Abschnitt wird am Beispiel der aus Kapitel 4 bekannten 133er BLU untersucht, wie sich die Anzahl der licht-beitragenden LEDs pro Pixel verhält, wenn ein Einfluss-Schwellwert  $L_{th}^{min}$  eingeführt wird, welcher bei Unterschreitung zur Reduktion des Einflusses auf 0 führt. Es gilt also

$$\forall p \in \mathcal{P}, l \in \mathcal{L} : \check{L}_{p,l} \leq L_{th}^{min} \rightarrow \check{L}_{p,l} := 0 \quad (5.5)$$

Der Einfluss wird quantisiert und mit einer Auflösung von 12 Bit gespeichert. Für verschiedene  $L_{th}^{min} \in \{0\%; 0,1\%; 0,5\%; 1,0\%\}$  ist in Abbildung 5.8a dargestellt wie sich pro Pixel die Anzahl der LEDs reduziert, wenn alle Einflüsse  $\check{L}_{p,l} \leq L_{th}^{min}$  nicht berücksichtigt werden. Es ist zu Erkennen, dass schon für mit  $L_{th}^{min} = 0\%$  nicht in jedem Pixel alle LEDs einen Beitrag leisten. Der gemessene Lichtbeitrag an manchen Positionen ist so gering, dass die 12 Bit Auflösung nicht ausreicht, um diese abzubilden. Andererseits sind in diesen Bereichen die Messwerte bei der Leuchtdichtemessung tendenziell mit Rauschen behaftet, wenn eine hohe Maximalleuchtdichte im Bild vorliegt. Somit ist dieser Bereich für sehr kleine Einflüsse vernachlässigbar. Entsprechend sind in Abbildung 5.8b die resultierenden Einflusssummen  $\sum_{l=0}^{N_L-1} \check{L}_l$  unter Berücksichtigung der Formel (5.5) dargestellt. Für den oberen Fall berechnet sich die Summe wie erwartet zu 1 in jedem Pixel. In Abbildung 5.9 sind die Häufigkeiten der Beitragssummen sowie der Anzahl der beitragenden LEDs für verschiedene  $L_{th}^{min}$  zu der Abbildung 5.8 aufgetragen. Die entsprechenden Median Werte und Standardabweichungen können im Anhang in Tabellen 5.2 und 5.3 nachgeschlagen werden. So ergibt sich für ein  $L_{th}^{min} = 0,5\%$  ein Median von 25 für die Anzahl der einflussnehmenden LEDs mit einer Standardabweichung von 3,92.

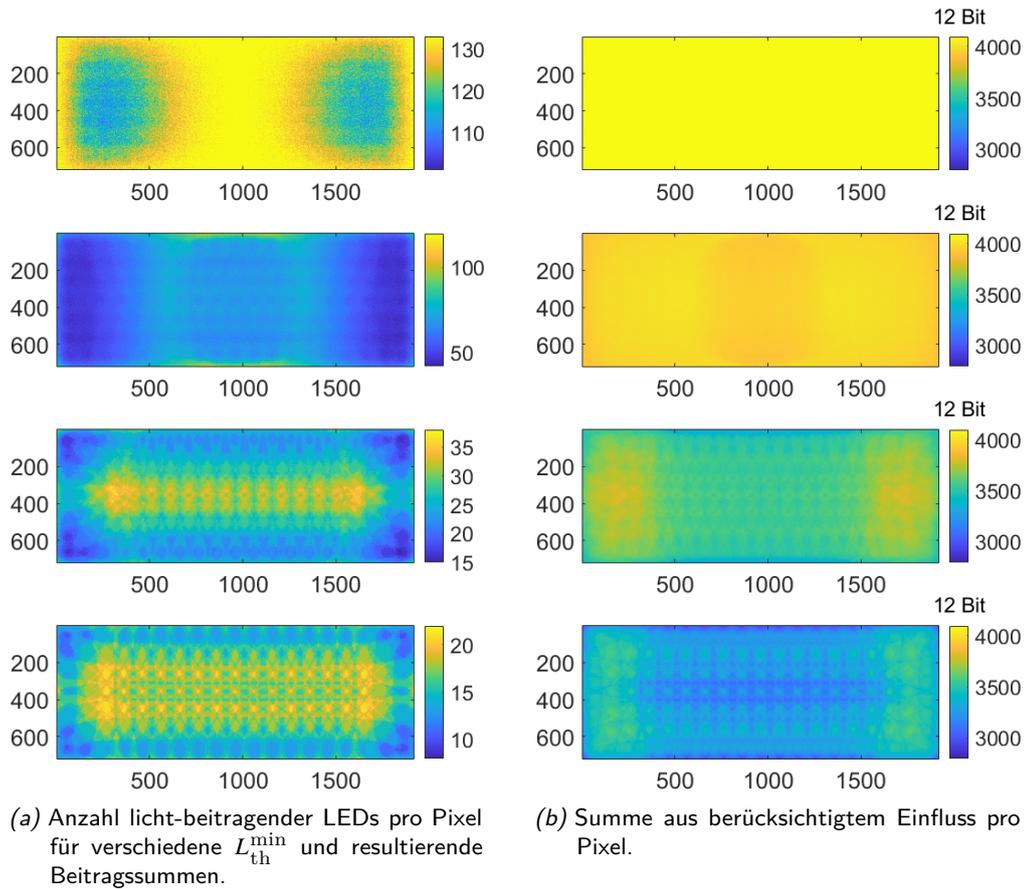


Abbildung 5.8.: Anzahl der LEDs mit Lichtbeitrag pro Pixel mit  $L_{th}^{min} \in \{0\%; 0,1\%; 0,5\%; 1,0\%\}$  von oben nach unten.

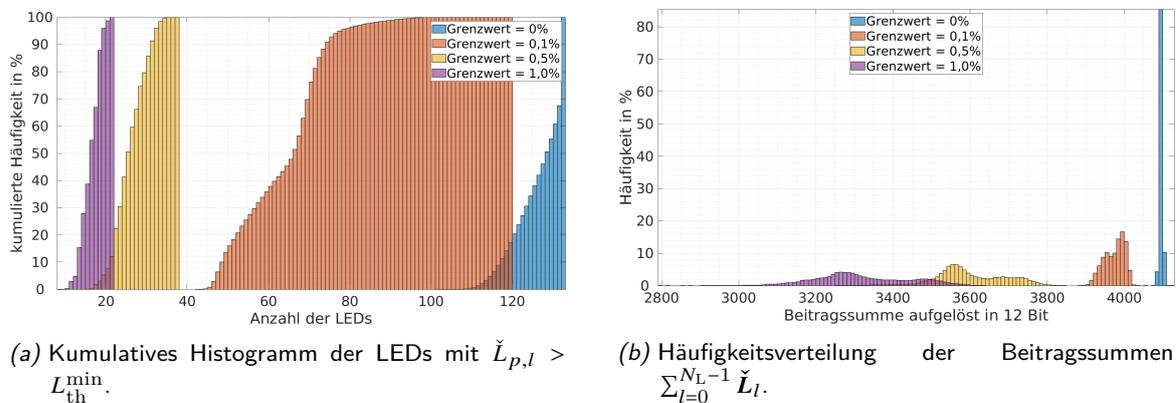


Abbildung 5.9.: Verteilungen der einflussnehmenden LEDs für verschiedene  $L_{th}^{min}$  sowie die entsprechende Verteilung der Beitragssummen.

### Auswirkung der Iterationszahl und betrachtete LEDs auf die Reduktion des Leistungsumsatzes

Die oben bereits genannte Abhängigkeit von der Iterationszahl  $N_{iter}$  wird in Abbildung 5.10 anhand des Powersavings des Saarschleifenbildes für eine BLU mit 240 LEDs dargestellt. Es handelt sich hierbei um die BLU, welche in Abbildung 4.14 (Seite 86) vorgestellt wurde. Das Powersaving auf der Ordinate ist

## 5.2. Konzept für eine effiziente Optimierung

	$L_{th}^{min}$			
	0 %	0,1 %	0,5 %	1,0 %
Median	128	66	25	16
STD	6,3009	10,2810	3,9196	2,2490

Tabelle 5.2.: Median und Standardabweichung der Anzahl an lichtbeitragenden LEDs basierend auf Abbildung 5.9. In 12 Bit Auflösung gespeichert.

	$L_{th}^{min}$			
	0 %	0,1 %	0,5 %	1,0 %
Median	4095	3979	3583	3301
STD	3,3607	26,1300	92,1740	122,8146

Tabelle 5.3.: Median und Standardabweichung der Beitragssummen in 12 Bit Auflösung.

gegen die  $\tilde{N}_L$  aufgetragen und für verschiedene  $N_{iter}$  visualisiert. Steigt die Anzahl der berücksichtigten LEDs, so steigt auch das Powersaving. Dieser Effekt rührt daher, dass mit steigenden  $\tilde{N}_L$ , mehr Information über den aktuellen Lichtbeitrag weiter entfernt liegenden LEDs zur Verfügung steht und die LED Erhöhung – je nach Beitrag – geringer ausfallen kann. Jedoch ist die Verbesserung des Powersavings ab einer bestimmten Anzahl  $\tilde{N}_L$  nicht mehr so hoch und die Kurven werden flacher. Weiterhin liefert eine kleine Iterationsanzahl ein geringeres Powersaving, da aufgrund des iterativen Charakters des SSC Algorithmus, die LEDs in den einzelnen Iterationen höher getrieben werden als das Optimum, da die Condenserwerte schon in der ersten Iteration zu hoch angenommen werden (fehlende Abstufungsmöglichkeit aufgrund zu geringer Iterationszahl) und folglich der Algorithmus die LED-Werte höher berechnet als theoretisch nötig. Beispielsweise liefert die Parameterkombination von  $\tilde{N}_L = 49$  und  $N_{iter} = 5$  etwa dieselbe Berechnungszeit

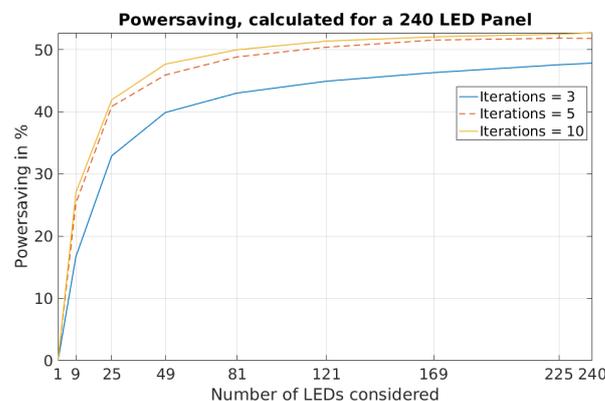


Abbildung 5.10.: Reduktion des Leistungsumsatzes für verschiedene  $\tilde{N}_L$  und bei verschiedenen ausgelegten Iterationszahlen  $N_{iter}$  der Pixelorder. Entnommen aus [Sch+17].

wie  $\tilde{N}_L = 25$  und  $N_{iter} = 10$ , jedoch ist das Powersaving für die letztere Kombination etwa 4 %-Punkte höher (entspricht einer Erhöhung um 9,5%). Die Kombination von  $\tilde{N}_L = 169$  und  $N_{iter} = 3$  liefert ungefähr dasselbe Powersaving wie  $\tilde{N}_L = 49$  und  $N_{iter} = 5$  jedoch mit einer doppelt so langen Berechnungszeit [Sch+17]. Somit stellt diese Parameterwahl immer einen Trade-Off dar, wie in Abschnitt 5.2.2 beschrieben

wurde.

In Abschnitt 4.5.2 wurde für die Analyse und Untersuchung der LSF bzw. BLU Eigenschaften ein konstanter Einfluss von 30 % angenommen, während das  $r_{50\%}$  Verhältnis variiert wurde. Wird nun das in dem aktuellen Kapitel vorgestellte Konzept mit der Berücksichtigung einer Teilmenge  $M_{i,j} \subset \mathcal{L}$  der LEDs angewandt, so berechnet sich die Reduktion des Leistungsumsatzes für ein automotive Testset für die ausgewählten LSF Modelle zu den in Abbildung 5.11 dargestellten Verläufen.

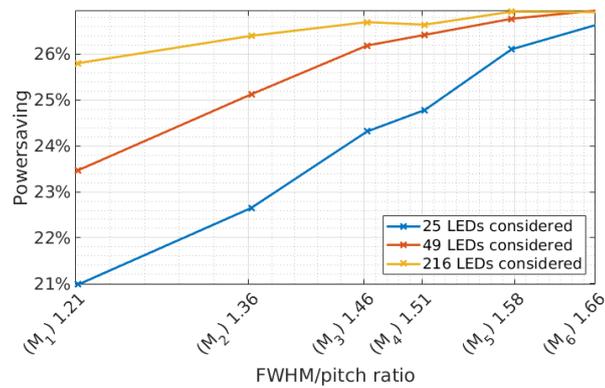


Abbildung 5.11.: Reduktion des Leistungsumsatzes bei konstantem  $\check{L}_{total}^{max}$  für verschiedene  $r_{50\%}$  und  $\check{N}_L$ . Entnommen aus [SRX19].

Auf der Abszisse ist das  $r_{50\%}$  Verhältnis aufgetragen während die Ordinate das Powersaving (Reduktion des Leistungsumsatzes) in Prozent darstellt. Die hier zugrunde liegende BLU besitzt  $N_L = 216$  LEDs. Somit entspricht die gelbe Kurve einem optimalen Ergebnis unter Berücksichtigung aller vorhandener LEDs, jedoch auch mit der höchsten Komplexität und Berechnungszeit. Die 216 LEDs wurden in der unteren Abbildung gewählt, da zum Zeitpunkt der Simulation die BLUs mit 216 LEDs aufgrund des Preises ein größeres Interesse erfuhren als die 240er. Qualitativ ist das Resultat jedoch ähnlich. Aus der Abbildung wird deutlich, dass für kleinere  $r_{50\%}$  das Powersaving bei gleichem maximalen Einfluss geringer wird. Insbesondere ist der Unterschied zwischen den verschiedenen  $\check{N}_L$  im unteren  $r_{50\%}$  Bereich größer, während er bei größeren  $r_{50\%}$  geringer ausfällt. Somit wird geschlussfolgert und empfohlen, dass  $r_{50\%}$  bei einem konstanten maximalen Einfluss größer gewählt werden soll. Dadurch kann die Komplexität ohne eine signifikante Verschlechterung des Powersavings reduziert und die Hardwarekosten gering gehalten werden [SRX19]. Somit kann der Local-Dimming Algorithmus in Hardware mit hoher Effizienz realisiert werden.

Mit diesem Konzept wird also ein Teil der Direct-Lit Herausforderungen gelöst und die Berechnungszeit limitiert, während die Lösung nah beim Optimum liegt.

### 5.3. Entwicklung eines LUT-basierten LSF-Modellkonzepts zur Speicherreduktion

Mit dem obigen Ansatz wird die zeitliche Limitierung umgangen und ermöglicht die Berechnung der Lösung innerhalb der gegebenen Constraints. Eine weitere Herausforderung betrifft den immer weiter steigenden Speicherbedarf für die LSF Information. An dieser Stelle wird auf Abbildung 5.5 (Seite 133) verwiesen. Weiterhin gilt die Limitierung des Speichers in Form von SRAM und Register. Steigende LEDs Zahlen können schnell den Speicherbedarf der Pixelorder in die Höhe treiben, sodass der Mehraufwand ökonomisch

nicht mehr gerechtfertigt werden kann. Der steigende SRAM Bedarf wurde bereits in Abbildung 5.6 auf Seite 134 in Abhängigkeit von  $\sqrt{N_L}$  dargestellt. Um diesen ungeachtet der steigenden LED Anzahl einzugrenzen, bedarf es eines weiteren neuartigen Konzeptes, welches in diesem Abschnitt vorgestellt und zur Nutzung in der Implementierung weiter angepasst wird.

#### 5.3.1. Idee und Konzept

Die vorgestellte Methode hat das Ziel, den stark gestiegenen Speicherbedarf, wie in Abschnitt 5.2 erläutert wurde zu reduzieren, um eine kostengünstige Hardwarelösung zu ermöglichen und weiterhin auf die Nutzung von einer *chipexternen* SRAM zu verzichten. Dabei wird die benötigte Datenmenge zur Speicherung der LSFs drastisch reduziert, indem die Einflüsse  $\check{L}_l$  der LEDs  $l \in \{0, \dots, N_L - 1\}$  nicht mehr für jede einzelne LED gespeichert, sondern nur noch durch eine geringe Anzahl an repräsentativen LSFs beschrieben werden. Da also eine Teilmenge ( $\mathcal{L}' \subset \mathcal{L}$ ) der vorhandenen LSFs zur Bestimmung aller Einflüsse genutzt wird, werden diese als *LSF-Modelle* bezeichnet.

Bei einer BLU, welche in Kapitel 4 beschrieben ist, sind die Leuchtdichteverteilungen der einzelnen LEDs ähnlich und ermöglichen die Nutzung des in den folgenden Abschnitten vorgestellten Konzeptes. Die Strahlengänge der einzelnen LEDs passieren den Stack im selben Abstand, sodass die Unterschiede gering sind. Dies gilt vor allem für LEDs welche von den Rändern weit entfernt sind. An den Rändern selbst, ist die Ähnlichkeit geringer, da hier zum einen das Licht von den Seitenwänden reflektiert werden kann und zum anderen die Leuchtdichte abnimmt, da hinter dem Rand keine weiteren LEDs vorhanden sind, welche zur Gesamtleuchtdichte beitragen können. Am Beispiel der 133er BLU werden die Unterschiede und Motivation bei der Wahl der repräsentativen LSFs verdeutlicht. Dazu zeigt Abbildung 5.12 die Positionen der einzelnen Kavitäten bzw. der innen-liegenden LEDs in Gelb dargestellt. Die zentral gelegene LED ist mit Magenta umrandet. Weiterhin deuten die 4 gestrichelten Linien die Querschnitte durch die einzelnen LSF, welche in Abbildung 5.13 dargestellt sind, an. Aufgrund der Symmetrie, werden nur die Querschnitte aus dem linken, oberen Quadranten visualisiert. Teil a) der Abbildung zeigt dabei die auf den jeweiligen Maximalwert normierten LSFs in einem einzelnen Plot dargestellt. Gut erkennbar ist, dass die Unterschiede zwischen den einzelnen Verläufen für diesen Fall nur marginal sind. In b) hingegen ist der resultierende Lichtbeitrag (*Einfluss*) nach Formel (2.15) aufgetragen.

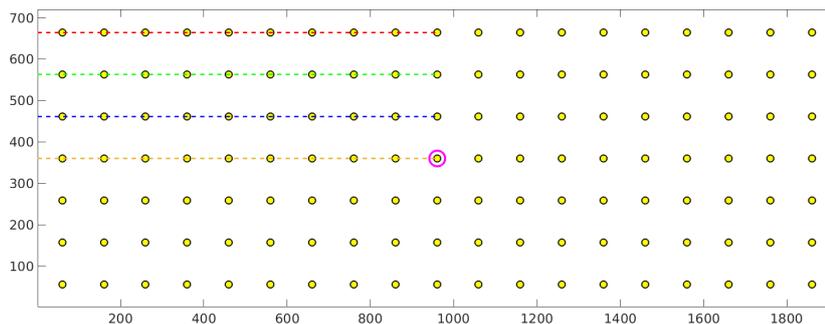


Abbildung 5.12.: Lage der LEDs der 133 BLU.

Diese Abbildung lässt einen erheblichen Unterschied zwischen den Einflüssen der einzelnen LSFs erkennen. Ebenfalls wird festgestellt, dass der maximale Einfluss jeweils bis zur dritten LED-Spalte um 20% abnimmt und schließlich konstant bleibt. Für die Querschnitte längs der Vertikalen ergibt sich ein hierzu analoges

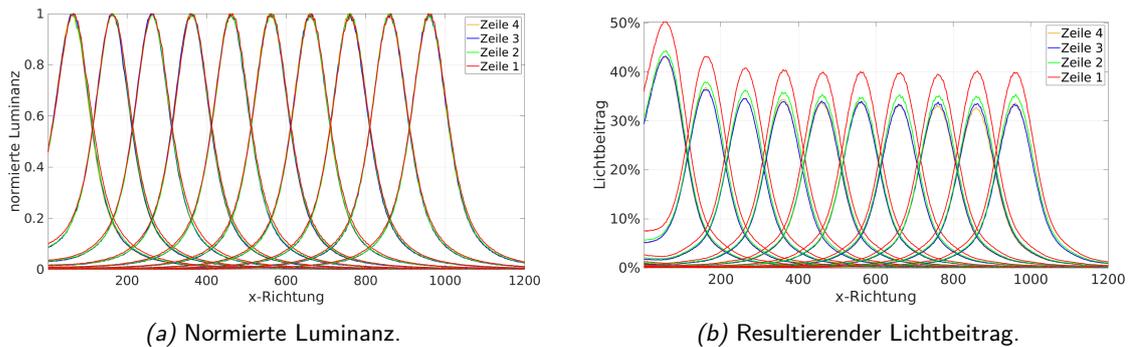


Abbildung 5.13.: Querschnitte durch die LSFs (Markierung aus Abbildung 5.12). Die Zeile 4 (Gelb) wird von Blau überlagert und nur in den Peaks sichtbar, da sie fast identisch sind.

Verhalten.

Durch die größeren Abweichungen (vgl. Abbildung 5.13b) müssen tendenziell mehr repräsentative Modelle verwendet werden. Die in der nachfolgenden Ausführung ausgewählten Modelle können in Hardware als ROMs in FPGAs oder ASICs realisiert werden.

Die Methodik zur Auswahl der Modelle muss Folgendes berücksichtigen

- Modellanzahl  $N'_L = |\mathcal{L}'|$  muss deutlich reduziert werden ( $N'_L \ll N_L$ )
- Konzept für beliebige Direct-Lit BLUs anwendbar
- Berechnungsaufwand in einer Pixelpipeline basierenden Architektur gering
- geringe Abweichung

Um eine sinnvolle Menge  $\mathcal{L}'$  zu bestimmen, wird ohne Beschränkung der Allgemeinheit die 133er BLU mit ihren LSFs auf das Modellierungsvorhaben untersucht. Zunächst werden dazu die Abweichungen der LSFs gegenüber einer Referenz berechnet wie im Folgenden dargestellt wird.

### Berechnung der Abweichung

Sollen nun LSFs miteinander auf Abweichung verglichen werden, müssen diese so *virtuell* verschoben werden, dass sich die LED Zentren überlagern. In diesem Fall kommt es jedoch zu Unstetigkeiten des verschobenen Skalarfeldes, da die Werte nur im Bereich der *Active Area* gemessen wurden, die Verschiebung jedoch eine Überlagerung mit nicht definierten Punkten induziert. Dieses Problem wird in Abbildung 5.14 visualisiert und verdeutlicht den Nachteil solch einer Methode. Die gezeigten Unstetigkeiten gilt es zu Vermeiden, da dadurch der berechnete Fehler ansteigt. In der genannten Abbildung wird beispielhaft eine zentral gelegene LSF  $L$  so verschoben, dass ein Vergleich mit  $L'$  möglich ist. Dabei treten die Sprünge aufgrund fehlender Messwerte auf, wie beispielhaft in dieser Abbildung dargestellt wird.

Eine Lösung für das Problem ist es, die gewählte Referenz zu fitten und so eine kontinuierlich definierte Funktion zu erzeugen, welche an jedem Punkt berechnet werden kann. Zur folgenden Bestimmung der quantitativen Abweichungen wird eine zentral gelegene LSF ausgesucht und mit einer Gauss-Superposition nach Formel (4.13) mit  $N = 3$  angenähert. Dazu wird als Basis der Querschnitt der ausgewählten Referenz genommen, da hier eine radial-invariante LSF vorliegt. Um den Fit symmetrisch zu halten, werden die 3 Mittelwerte  $\mu_n$  auf einen konstanten Wert forciert. Der resultierende Fit kann dann an beliebigen Koordinaten

### 5.3. Entwicklung eines LUT-basierten LSF-Modellkonzepts zur Speicherreduktion

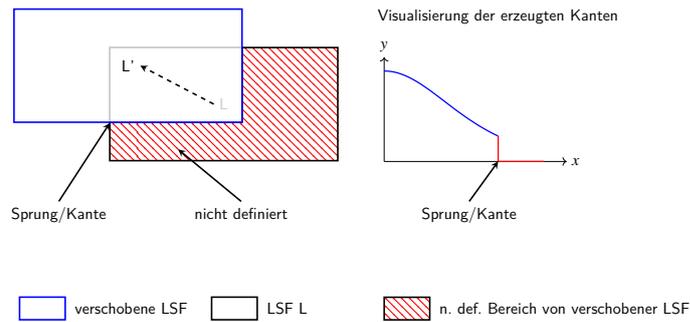


Abbildung 5.14.: Verdeutlichung des Problems beim Vergleich zwischen LSFs an verschiedenen Positionen.

zur Berechnung einer Referenz-LSF genutzt werden. Der resultierende Fit ist zusammen mit dem Querschnitt der Messdaten der mittleren LSF in Abbildung 5.15 dargestellt.

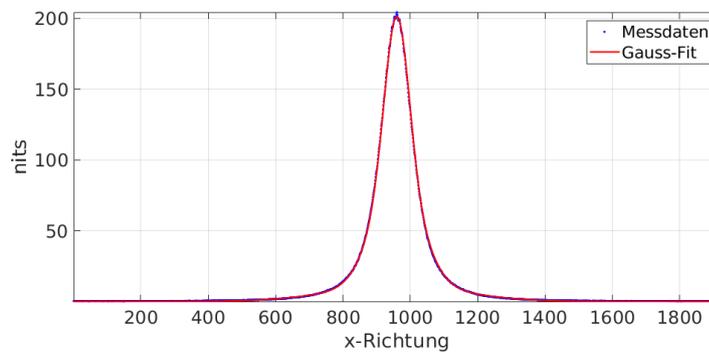


Abbildung 5.15.: Messdaten und resultierender Fit einer Gauss-Superposition ( $R^2 = 0,9998$ ).

Wird nun der RMSE zwischen jeder LSF (als Einfluss, wie in Abbildung 5.13b) und der vorgestellten Referenz durch Verschiebung auf die LED-Zentren berechnet, so ergibt sich die in Abbildung 5.16 visualisierte *Heatmap*, welche für jede LED den RMSE zur Referenz zuordnet und farblich kennzeichnet. Zusätzlich sind die y- bzw. die x- Symmetrieachsen in Weiß mit gestrichelten Linien eingezeichnet. Diese Symmetrie wird auch durch die resultierenden Fehler deutlich und entspricht der Erwartung. Die leichten Unterschiede kommen durch die Messungenauigkeiten zustande und können vernachlässigt werden. An dieser Stelle wird nochmal darauf hingewiesen, dass der Algorithmus auch mit leichten Abweichungen von den tatsächlichen Messungen funktionieren und somit eine gewisse Robustheit aufweisen muss. Aber auch die LSF selbst hat einen Einfluss auf die möglichen Artefakte, welche basierend auf ungenauen LSF Messungen entstehen können. Dazu wird auf Abschnitt 4.5 *Auswirkungen der LSF-Form auf die Robustheit und Qualität* verwiesen.

Anhand dieser Fehler-Heatmap können nun Bereiche ähnlicher *Abweichung/RMSE* klassifiziert werden. Diese sind in der Abbildung jeweils mit einem roten Rechteck markiert, wobei nur der linke obere Quadrant aufgrund der erkannten Symmetrie von Interesse ist. Diese Bereiche sind aufsteigend nummeriert. Insbesondere berechnet sich zu den Einflüssen aus Bereich 10 nur ein geringer Fehler.

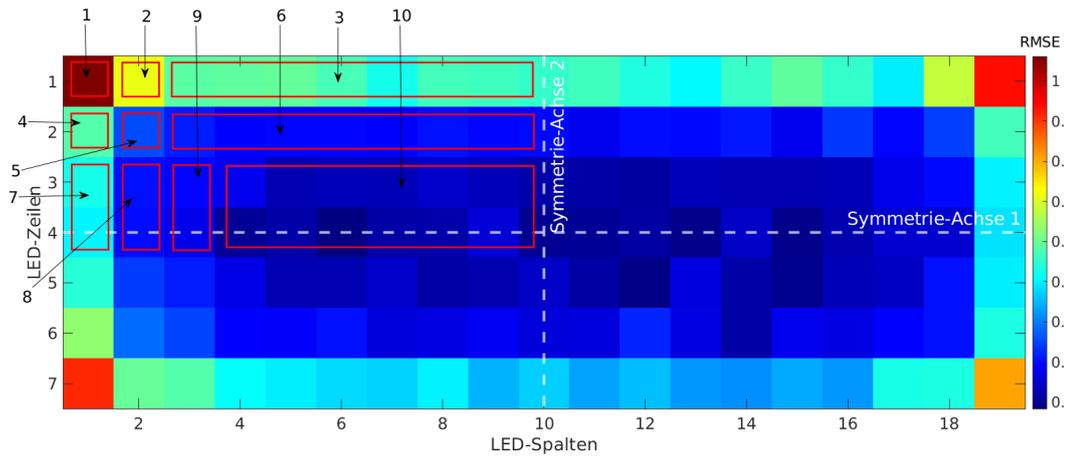


Abbildung 5.16.: Resultierender Fehler zum Fit aus Abbildung 5.15 (RMSE). Die Skala ist einheitenlos, da der RMSE zwischen relativen Werten im Bereich  $0 \dots 1$  in der Fläche berechnet wird. 1 entspricht 100 %.

### Wahl der Modelle und Konzept

Basierend der oben gezeigten Abbildung kann nun eine sinnvolle Modellauswahl durchgeführt werden. Dabei soll die Symmetrie-Eigenschaft zur Anwendung kommen und ausgenutzt werden. Wird Bereich 9 und 10 aus Abbildung 5.16 zusammengefasst, entstehen 9 einzelne Bereiche mit verschiedenen hohen Abweichungen untereinander. Somit können sie jeweils durch ein eigenes Modell repräsentiert werden. In diesem Fall ist  $|\mathcal{L}'| = 9$ . Ein entsprechendes Modell mit linearem Index  $i$  wird abstrakt mit

$$M'_i \in \mathcal{L}', \forall i \in \{1, \dots, N'_L\}$$

und das zugehörige Skalarfeld analog zu Formel (2.15) (Seite 33) mit

$$M'_i \in [0; 1]^{N'_F \times N'_P}$$

beschrieben. Die resultierende Auswahl ist in Abbildung 5.17 in Rot dargestellt. Es handelt sich hierbei – unabhängig von der LED-Anzahl – um die  $3 \times 3$  LEDs in der oberen linken Ecke des Displays, da die Symmetrie ausgenutzt werden kann. Aufgrund der hohen Abweichung im Vergleich zu den übrigen, werden die Modelle 1, 2 sowie 4 und 5 nur an ihren festen Positionen genutzt, sowie aus dem symmetrischen Aufbau resultierenden Spiegelungen an x- und y-Achse. Die übrigen Modelle 3, 6, 7, 8 und 9 werden hingegen wie in der Abbildung dargestellt zur Repräsentation der weiteren LSFs verwendet (grün, gestrichelte Pfeile). Diese sind in der Abbildung blau umrandet. Zur Verdeutlichung der Spiegelung bzw. der Verwendung der Modelle sind beispielhaft ausgewählte Positionen mit hellrot dargestellt und zeigen die zugehörige Modellnummer mit einem Strich (z.B.  $1 \rightarrow 1'$ ). Sollen diese vorgeschlagenen Modelle für den Local-Dimming Prozessor nun genutzt werden, so muss noch das bereits in Abbildung 5.14 visualisierte Problem der virtuellen *Verschiebung* der LSFs behoben werden, um die Modelle wie vorgeschlagen nutzen zu können. Es wird schnell ersichtlich, dass beispielsweise die Nutzung des Modells 9 an einer anderen als der ursprünglichen LED Position, das Problem der *fehlenden* Werte erzeugt und folglich eine Zuordnung des Einflusses abseits des Messfeldes nicht möglich ist.



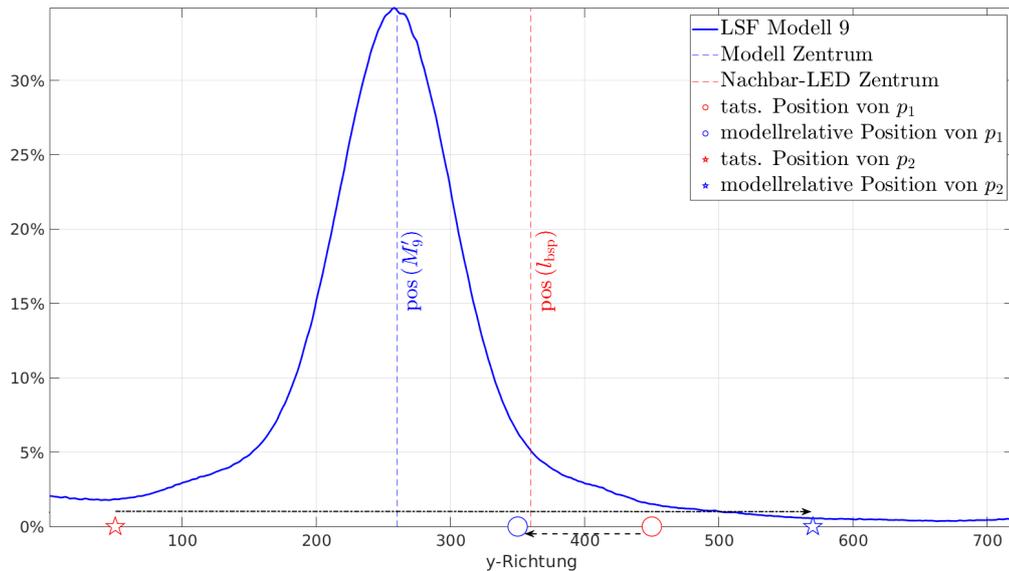


Abbildung 5.19.: Beispiel zur Bestimmung der virtuellen Position bezogen auf ein Modell in Abhängigkeit der Pixellage.

Formel (4.1) lässt sich der lineare LED Index  $l_{bsp}$  für dieses Beispiel berechnen. Die Funktion  $pos(\cdot)_y := pos(\cdot)$  liefere die entsprechende Koordinate. Somit bestimmt sich der Einfluss der LED  $l_{bsp}$  in einem Pixel  $p$  mit Position  $pos(p)$  zu

$$\check{L}_{p, l_{bsp}} = \begin{cases} M'_9 (pos(M'_9) + pos(p) - pos(l_{bsp})), & pos(p) - pos(l_{bsp}) > 0 \\ M'_9 (pos(M'_9) + pos(l_{bsp}) - pos(p)), & \text{sonst} \end{cases} \quad (5.6)$$

Die schwarzen Pfeile deuten in der Abbildung den Positionswechsel von der tatsächlichen Pixelposition auf die entsprechende modellbezogene Position. Das Beispiel von  $p_2$  (Pentagon) veranschaulicht das Prinzip der Spiegelung des Modells wie in Abbildung 5.18 dargestellt.

Für die spätere Umsetzung müssen somit *Mapping*-Funktionen eingeführt werden um die Ausrichtung und Zugehörigkeit zwischen LEDs und Modelle zu bestimmen. Folgende Eigenschaften sollen dabei erfüllt werden

- Jeder LED $_l$  wird ein Modell aus  $\mathcal{L}'$  zugeordnet
- Für jede LED muss bekannt sein, ob das zugehörige Modell an x- oder y-Achse gespiegelt werden muss (Ausrichtung)
- Zusätzlich muss bekannt sein, ob sich die LED bzw. das zugehörige Modell an einem der Ränder befinden (Lage, siehe Problem aus Abbildung 5.18)

Die Modell-Mapping-Funktion  $map_{LED}^{Modell} : \mathcal{L} \rightarrow \mathcal{L}'$  kann überführt werden in eine äquivalente Funktion, welche eine Zuordnung der LED-Nummer durchführt

$$map_{LED}^{Modell} : \{1, \dots, N_L\} \rightarrow \{1, \dots, N'_L\}$$

Analog dazu, wird die Modell-Ausrichtungsfunktion wie folgt definiert

$$\text{map}_{LED}^A : \{1, \dots, N_L\} \rightarrow \{0, \dots, 3\}$$

$$l \mapsto \begin{cases} 1, & \text{gespiegelt an y-Achse} \\ 2, & \text{gespiegelt an x-Achse} \\ 3, & \text{gespiegelt an x- und y-Achse} \\ 0, & \text{sonst} \end{cases}$$

Für einige der Modelle ist es ebenso notwendig zu wissen, ob sich die zu modellierende LED an einem der Ränder befindet, oder verschoben wurde. Diese Modell-Rand-Funktion definiert sich zu

$$\text{map}_{LED}^R : \{1, \dots, N_L\} \rightarrow \{0, \dots, 3\}$$

$$l \mapsto \begin{cases} 1, & \text{oberer Rand} \\ 2, & \text{linker Rand} \\ 3, & \text{linker und oberer Rand} \\ 0, & \text{sonst (kein Rand)} \end{cases}$$

Das Konzept basiert also auf der Zuordnung eines repräsentativen, gespeicherten Modells zu einer LED. Für die spätere Realisierung werden folglich die eingeführten Mapping-Funktionen benötigt. Am Beispiel der 240er BLU sind diese Zuordnungen in Abbildung 5.20 visualisiert.

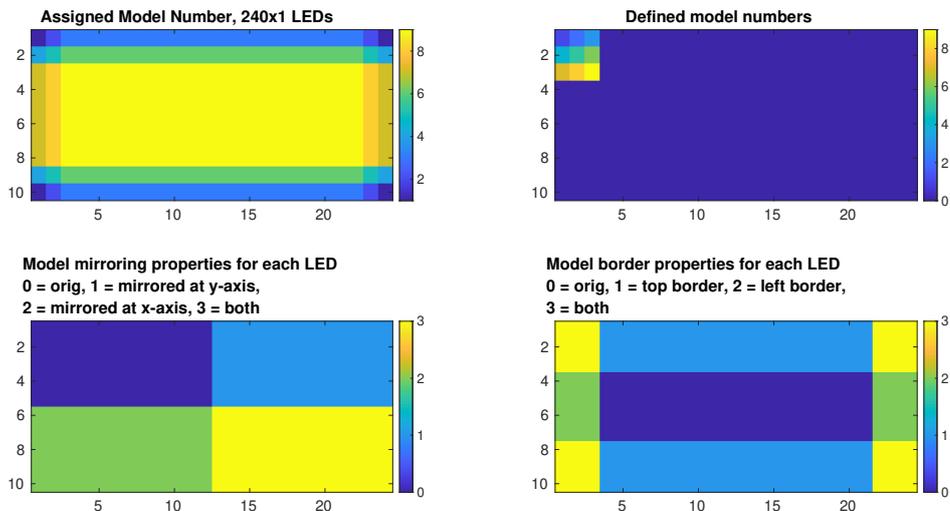


Abbildung 5.20.: Visualisierung der Mapping-Funktionen am Beispiel der 240er BLU.

### 5.3.2. Verfahren zur konsistenten Modellierung

In den vorherigen Abschnitten wurde das Konzept der Modellauswahl eingeführt. In diesem Abschnitt wird für diese Methode ein mathematisch konsistentes Verfahren vorgestellt, um die Modelle zu generieren. Wie weiter oben beschrieben, sind die Modelle so gewählt, dass die Symmetrie ausgenutzt werden kann.

Jedoch kommt es aufgrund von Produktionsschwankungen und auch inhomogene mechanische Belastung des LC-Panels zu einer Imperfektion des Backlights. Das All-On weicht von einer homogenen Fläche oft ab. Wegen  $|\mathcal{L}'| \ll |\mathcal{L}|$  unterscheidet sich das modellbasierende All-On zudem vom gemessenen All-On aufgrund der Systematik. Somit werden in sich konsistente Modelle benötigt.

Dazu wird eine symmetrische Lage der LEDs innerhalb der BLU vorausgesetzt und idealisiert. Folglich wird ein symmetrisches All-On bezüglich der vertikalen bzw. der horizontalen Mittellinien angenommen. Die in der Realität vorkommenden Abweichungen der Uniformität werden somit nicht in das Modell übernommen. In Kapitel 4 wurde bereits gezeigt, dass gutes BLU-Design hilfreich für die Robustheit des Systems bezüglich der Fehleranfälligkeit ist.

In Abbildung 5.21 wird das Verfahren zur Erzeugung der Modelle in Form eines Diagramms dargestellt und lässt sich in zwei Phasen unterteilen.

### Phase 1: Modellfindung

In der ersten Phase werden die Modelle ausgewählt. Dies wird nur einmalig für ähnliche BLUs durchgeführt. Folglich müssen auch in dieser Phase zur erstmaligen Modellauswahl alle  $N_L^c \cdot N_L^r$  LSFs zuzüglich einer All-On Messung in ihrer Leuchtdichteverteilung gemessen werden. Unter Ausnutzung der symmetrischen LED-Positionen, kann die Anzahl der Messungen bereits in dieser Phase auf  $\lfloor (N_L^c + 1)/2 \rfloor \cdot \lfloor (N_L^r + 1)/2 \rfloor + 1$  reduziert werden. Diese Messungen werden auch zur Abdeckung der fehlenden LSFs verwendet. Die Auswahl der Modelle  $\mathcal{L}'$  erfolgt wie in Abschnitt 5.3.1 vorgestellt. Für diese Arbeit seien die Modelle wie oben festgelegt und argumentiert wurde gewählt, dabei ist  $N_L' = 9$ .

### Phase 2: Konsistente Modellberechnung

In der zweiten Phase geht es um die in sich konsistente Berechnung des zu modellierten Einflusses für die in Phase 1 ausgewählten Modelle. Werden die in der vorherigen Phase gewählte repräsentative LED-Positionen beibehalten ( $3 \times 3$  LEDs in der oberen linken Ecke), so werden pro BLU, unabhängig der LED-Anzahl  $N_L$ , genau  $N_L'$  Messungen benötigt. Zusätzlich müssen die LED Koordinaten  $\text{pos}(\text{LED}_i) \equiv \text{pos}(i), \forall \text{LED}_i \in \mathcal{L}$  (in Pixelauflösung) bekannt sein bzw. aus den Messungen extrahiert werden. Bei LSFs mit einem globalen Maximum befindet sich das LED-Zentrum an der Stelle des gefundenen Maximums. In dieser Phase werden auch die eingeführten Mapping-Funktionen  $\text{map}_{\text{LED}}^{\text{Modell}}(i)$ ,  $\text{map}_{\text{LED}}^{\text{A}}(i)$  und  $\text{map}_{\text{LED}}^{\text{R}}(i)$  generiert (Beispiel in Abbildung 5.20).

Die Methodik zur Berechnung der Modelleinflüsse läuft nach folgendem Schema ab.

1. Leuchtdichtemessung der 9 Modelle ( $M_i^{\text{abs}}, i \in \{1, \dots, 9\}$ )
2. Berechnung der oben genannten Mapping-Funktionen
3. Bestimmung der LED-Zentren
4. Berechnung eines *synthetischen* All-Ons ( $L'^{\text{abs}}$ ) aus den 9 Messungen und LED-Positionen
5. Normierung analog zu Formel (2.15) der gemessenen Modelle mithilfe des synthetischen All-Ons zu  $M_i'$

Dieses neuartige Verfahren liefert mathematisch konsistente Einflussmodelle und ist nicht auf eine festgelegte Modellanzahl  $N_L'$  beschränkt, sondern generell anwendbar.

Zur Verdeutlichung des Verfahrens werden in den folgenden Abbildungen Beispiele zu den Schritten abgebildet.

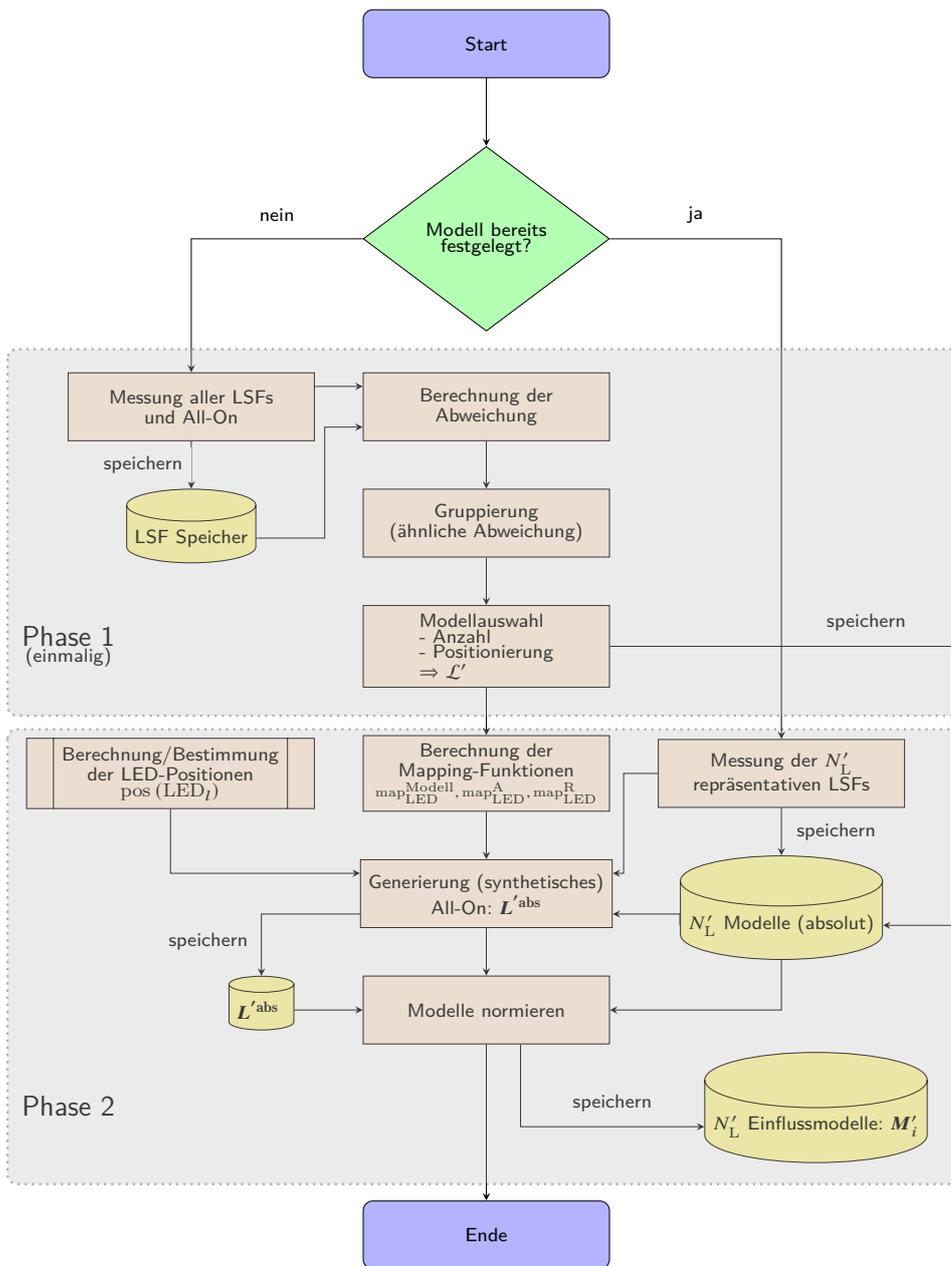
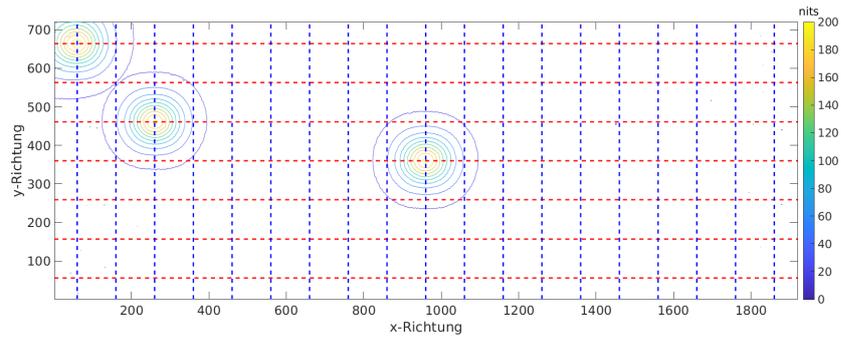


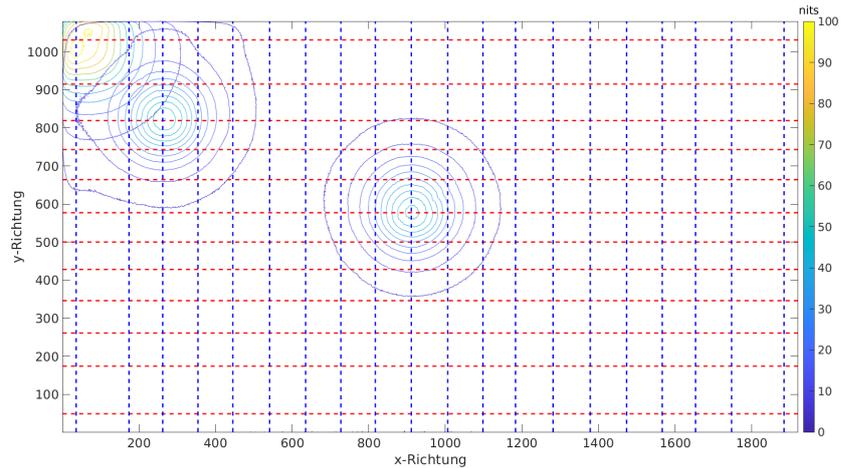
Abbildung 5.21.: Ablauf zur Erzeugung konsistenter Modelle.

So ist in Abbildung 5.22 die Lage der LEDs bzw. deren Bestimmung visualisiert. Dabei liegen die LED-Zentren an den Schnittpunkten der roten und blauen Linien. Beispielhaft sind die Höhenlinien ausgewählter LSF zur Verdeutlichung der Lage eingezeichnet. In Abbildung 5.22a ist dabei eine 133er BLU eines ICDs dargestellt (Seitenverhältnis 8:3), während in b) die Positionen der LEDs eines Consumer-TV-Gerätes mit einem Seitenverhältnis von 16:9 abgebildet wird. Auffallend sind dabei die nicht äquidistanten Abstände der äußeren LED-Zeilen bzw. Spalten zu den übrigen. Diese größeren Abstände sind mit dem Gedanken einer höheren Uniformität realisiert und können mit beispielsweise 9 Modellen abgedeckt werden.

Der nächste Schritt, die Generierung eines synthetischen All-Ons aus 9 Messungen der 133er BLU ist in



(a) LED-Positionen einer 133er BLU eines ICDs.



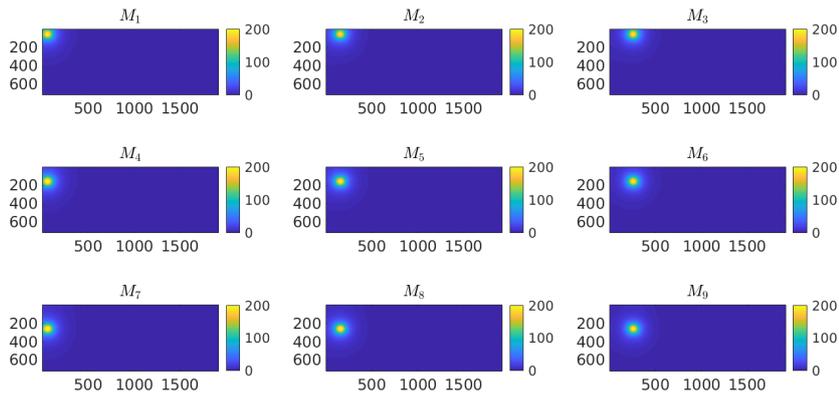
(b) LED-Positionen der 240er BLU eines Consumer-Gerätes.

Abbildung 5.22.: Beispiel der Bestimmung von LED-Zentren mit 3 beispielhaft ausgewählten LSFs.

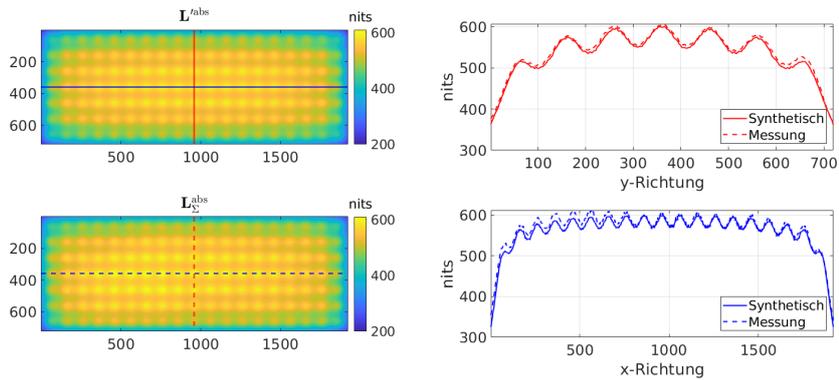
Abbildung 5.23 dargestellt (in Pixelauflösung). Dabei sind in a) die Messungen ( $M_i^{\text{abs}}, i \in \{1, \dots, 9\}$ ) und in b) das resultierende synthetische All-On ( $L^{\text{abs}}$ ) im Vergleich mit dem tatsächlich gemessenen Skalarfeld  $L_{\Sigma, l}^{\text{abs}} \Big|_{l=1}$  dargestellt. Hierzu werden die Mapping-Funktionen und LED-Koordinaten ( $\text{pos}(l), l \in \{1, \dots, N_L\}$ ) genutzt und analog zu dem Beispiel in Formel (5.6) die Leuchtdichte in jedem Pixel  $p$  berechnet. Das Skalarfeld wird also durch geometrische Verschiebung und Spiegelung der Messungen erzeugt. In Abbildung 5.23b sind auf der rechten Seite zur Verdeutlichung die entsprechenden Querschnitte dargestellt. Die gestrichelten Linien zeigen dabei den Schnitt durch das gemessene All-On, während die durchgezogene Linie das generierte All-On repräsentiert. Insgesamt ergibt sich nur eine geringe Abweichung. Zusätzlich ist ein positiver Effekt erkennbar. Die suboptimalen Messungen (e.g. Amplitudenunterschiede wie im horizontalen Schnitt erkennbar) können durch das Verfahren korrigiert werden, da die Modelle auch im All-On eine symmetrische Leuchtdichte induzieren. Zusätzlich werden die 9 Messungen noch einer Filterprozedur unterzogen, um die Ausreißer der Messungen zu unterdrücken.

Basierend auf diesem synthetischen All-On werden die 9 ausgewählten Messungen normiert, sodass die resultierenden Modelle  $M_i', \forall i \in \{1, \dots, 9\}$  den *Einfluss* in einem Pixel repräsentieren. Die Normierung erfolgt analog zu Formel (2.15) auf Seite 33. Wird der Operator  $\otimes$  zur Berechnung der elementweisen

### 5.3. Entwicklung eines LUT-basierten LSF-Modellkonzepts zur Speicherreduktion



(a) 9 Leuchtdichtemessungen der ausgewählten LSFs (Panelauflösung). Skala in  $\text{cd m}^{-2}$ .



(b) Synthetisches All-On ( $L'^{\text{abs}}$ ) generiert aus den Messungen aus a) im Vergleich mit realem All-On  $L_{\Sigma,t=1}^{\text{abs}}$  einer Leuchtdichtemessung mit zugehörigen Querschnitten.

Abbildung 5.23.: Vergleich des synthetischen All-Ons aus den 9 Messungen basierend auf einer 133 BLU.

Division zweier Matrizen verwendet, so gilt

$$M'_i = L_l^{\text{abs}} \oslash L'^{\text{abs}}, \forall i \in \{1, \dots, 9\} \quad (5.7)$$

Dabei ist die LED-Nummer  $l$  abhängig von der Modellnummer  $i$  und wird durch eine LUT realisiert. Für die 133er BLU sieht die Zuordnung wie in Tabelle 5.4 dargestellt aus (Nummerierung der LED-Nummern analog zu Abbildung 4.3, Seite 78). Jedes Modell (mit Nummer  $i$ ) wird durch eine LSF einer LED (mit Nummer  $l$ ) repräsentiert. Die resultierenden Einflussmodelle sind in Abbildung 5.24 abgebildet. Je weiter sich die Modelle vom Rand befinden, desto geringer ist der maximale Einfluss. Dies wurde in den Querschnitten aus Abbildung 5.13b bereits ersichtlich.

Modellnummer $i$	1	2	3	4	5	6	7	8	9
LED-Nummer $l$	1	2	3	20	21	22	39	40	41

Tabelle 5.4.: Zuordnung der Modell- zu LED-Nummern.

Diese Einflussmodelle können für die Repräsentierung des Einflusses jeder einzelnen LED ( $\check{L}'_i$ ) genutzt

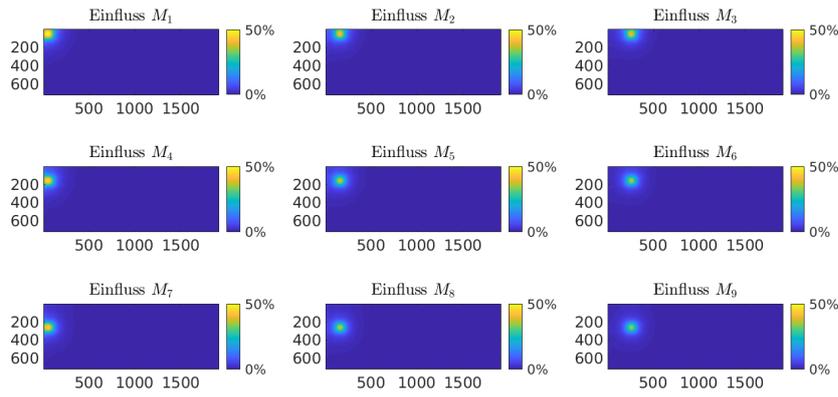


Abbildung 5.24.: Skalarfelder  $M'_i$  der Modelle 1...9 resultierend aus der Normierung mit synthetischem All-On (Panelauflösung).

werden. Dazu wird zur Verdeutlichung des Verhaltens die Summe der Einflüsse basierend auf den Modellen gebildet und wird in Abbildung 5.25 dargestellt. Zu erkennen ist, dass die Summe von  $\mathbb{1}^{N_P^E \times N_P^C}$  abweicht. Besonders gut wird das Resultat in Bereichen der originalen Positionen der Modelle sowie deren Spiegelungen. Dieser Umstand muss während der Berechnung bzw. bei der Pixelkompensation berücksichtigt werden. In diesem Fall errechnet sich ein RMSE von 0,0177. Trotz der Ungleichheit zu  $\mathbb{1}^{N_P^E \times N_P^C}$  resultiert dieses

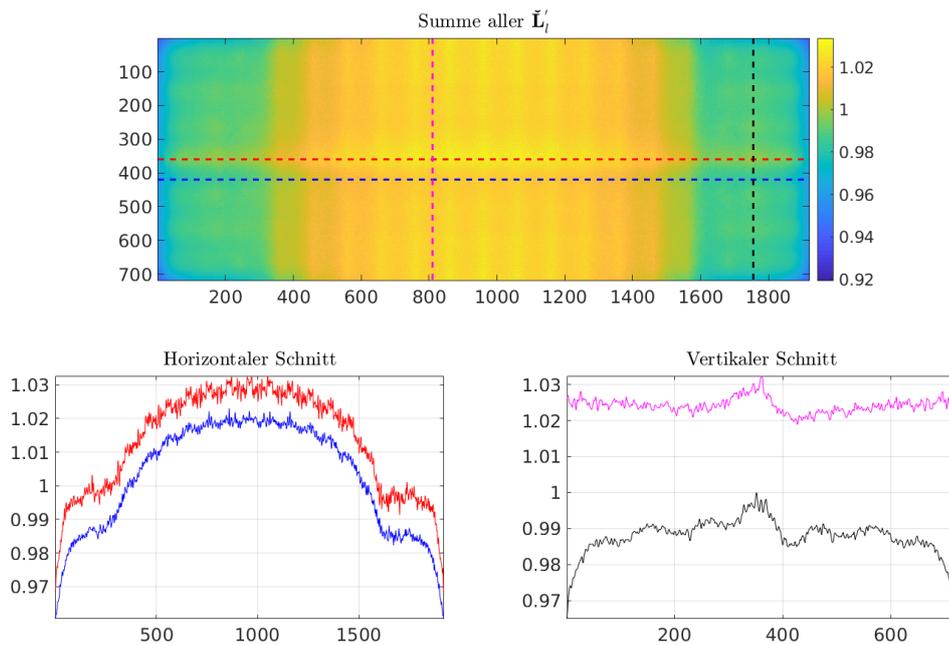


Abbildung 5.25.: Modellbasierende Summe der Einflüsse in jedem Pixel.

Verfahren in einem in sich konsistenten Resultat.

Somit wird deutlich, dass die konsistente Modellbildung sinnvoll und hilfreich ist. Die Abweichungen zur idealen Einsmatrix werden dadurch reduziert. Diese ergibt sich, wenn die LSFs für alle LEDs nach Formeln (2.15) und (2.16) ohne Modelle berechnet werden.

### 5.3.3. Reduzierung auf Sample-Points

Bisher basierte die Modellierungsmethode in der vollaufgelösten nativen Panelauflösung. Jedoch ist das Ziel, das Verfahren direkt auf einer Pixelpipeline in Hardware zu realisieren und verwenden. Folglich würden diese 9 ausgewählten Modelle mit einer Quantisierung von 12 Bit einen Speicher von 149,2992 MBit benötigen (bei einer Auflösung von  $1920 \times 720$ ) und den vorhandenen Block-RAM im verwendeten FPGA um den Faktor 10 überschreiten. Aufgrund der Kosten und Echtzeitfähigkeit, soll wie bereits in den vorangegangenen Kapiteln erwähnt, kein externer Speicher genutzt werden. Somit müssen die Modelle in ihrer Auflösung in einer reduzierten Form gespeichert werden. Für die spätere Pixelkompensation wird das resultierende Licht hinter jedem Pixel in Echtzeit benötigt. Um diese zu realisieren wird das aus den LED-Werten und LSF erzeugte Backlight *virtuell* mit einer geringeren Auflösung gespeichert (Stützstellen, auch *Sample-Points*) und wird wie beim SoA in der Pixelpipeline interpoliert. Erlaubt wird die Reduktion ebenfalls durch die Tatsache, dass die räumliche Frequenz der LSF viel geringer ist als die Pixelfrequenz. Eine logische Folge ist, die Sample-Points zu der Auflösung der Modelle passen müssen, um die Effektivität hoch und den Fehler kleinzuhalten.

Es wurde bereits argumentiert, dass nach dem Verfahren des SoA mit der identischen Anzahl von Condenserezellen und Sample-Points ( $N_C = N_{SP}$ ) maximale Anzahl an LEDs und auch der Speicher (SRAM) schnell limitiert werden (siehe auch Abschnitt 5.1.2), die Anzahl aber auch nicht beliebig gewählt werden kann (siehe Abschnitt 5.1.1). Es liegt also Nahe, die Bindung der Zellenanzahl aufzuheben um einen größeren Optimierungshebel ansetzen zu können. Für die Anzahl der Sample-Points wird für ein Seitenverhältnis von 8:3 für die Horizontale ein  $N_{SP}^c = 129$  sowie für die Vertikale ein  $N_{SP}^r = 49$  gewählt und entspricht einer Gesamtanzahl von  $N_{SP} = 6321$ . Die Anzahl entspricht der doppelten Anzahl Condenserzeilen sowie -spalten unter Berücksichtigung der äußersten Bildgrenzen. Weiter unten wird die Lage beschrieben. Daraus ergibt sich ein SRAM Bedarf von 682,668 kBit und entspricht nur noch 4,5 % der voll-aufgelösten Modelle.

#### Anordnung der Sample-Points und Condenserezellen

Zunächst wird die Anordnung und die Lage der Sample-Points in Relation zu den Condenserezellen definiert. Da in der Pixelkompensation die berechneten Werte des Backlights zwischen den Stützstellen bilinear interpoliert werden müssen, werden die SPs so platziert, dass die äußersten Zeilen und Spalten des Bildes abgedeckt werden. Zudem ist die Anzahl der Condenserezellen für eine korrekte und effiziente Berechnung so zu wählen, dass zusätzlich zum Powersaving, die Grenzen der Zellen zu den Sample-Points hin ausgerichtet sind, sodass die Zuordnung der Condenserkoordinaten zu SP-Koordinaten einfach durchgeführt werden kann. Des Weiteren soll das Zentrum einer Zelle ebenfalls auf einem Sample-Point liegen. Im Rahmen dieser Arbeit berechnet sich die Anzahl der Condenserezellen zu

$$N_C = \frac{N_{SP}^c - 1}{2} \cdot \frac{N_{SP}^r - 1}{2} = 1536$$

sowie

$$N_C^c = 64 \text{ und } N_C^r = 24$$

Dazu ist in Abbildung 5.26 Struktur sowie die Anordnung der Zellen und Sample-Points dargestellt. Die roten Quadrate repräsentieren hierbei die Lage der einzelnen Sample-Points innerhalb der *Active-Area* des Displays. Der Abstand beträgt 15 Pixel in der Horizontalen als auch in der Vertikalen. Somit repräsentiert ein Sample-Point eine Fläche von  $15 \times 15$  Pixel und ist beispielhaft mit als hellrote Fläche markiert. Die

Lage eines Sample-Point wird durch ihre Spalte und Zeile ( $SP_{r,c}$ ) beschrieben. Ebenso ist ein linearer Index möglich. Die Pixel innerhalb dieser Fläche wird mit  $p_{i,j}^{SP}$  dargestellt.

Analog zu den Sample-Points, ist eine beispielhaft ausgewählte Condensierzelle ausgewählt und die Lage mit einer grünen Fläche markiert. Es ist ersichtlich, dass die Dimensionen der x- und y-Richtungen hier genau doppelt so viele Pixel abdecken ( $30 \times 30$ ). Die Position der Zelle  $C$  wird analog zu den Sample-Points mit Zeilen und Spalten oder linear indiziert angegeben. Die in einer Zelle befindlichen Pixel werden bei Bedarf ebenfalls über ihre lokalen (zellenbezogene) Koordinaten adressiert ( $p_{i,j}^C$ ).

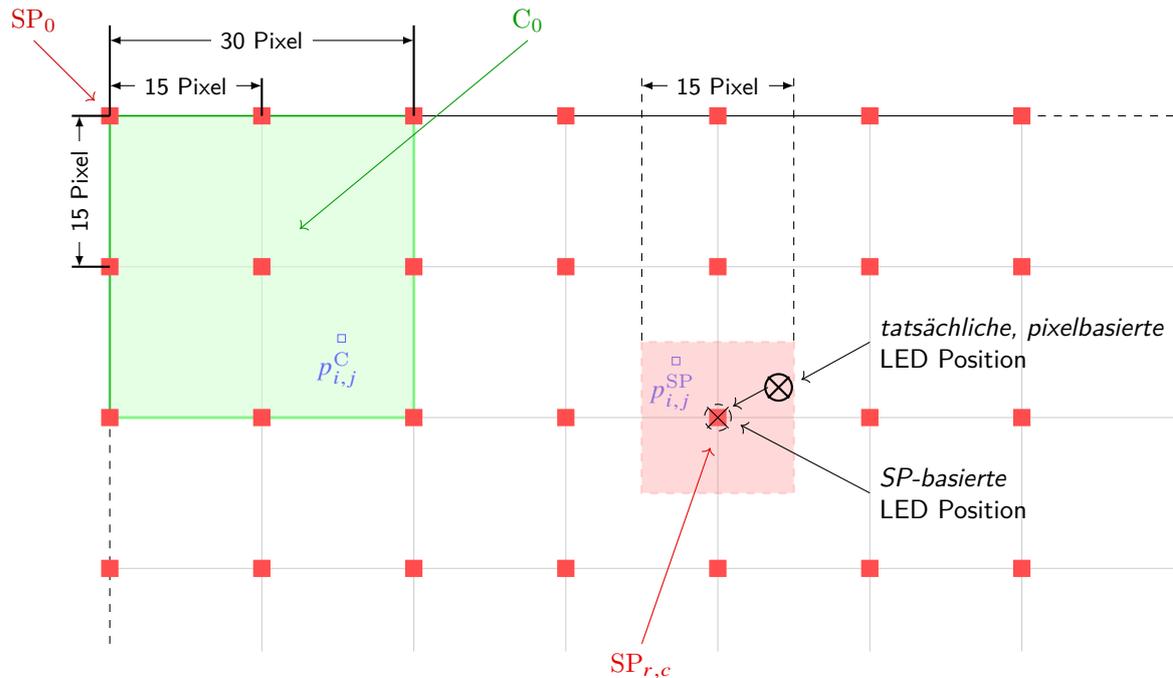


Abbildung 5.26.: Aufbau und Lage, sowie Relation der Condensierzellen sowie der Sample-Points.

## LED-Positionen

In der oberen Abbildung ist ebenfalls dargestellt, wie die tatsächliche LED-Position in Pixelauflösung von einer Sample-Point Position abweichen kann. Die betrachtete LED mit ihrem Zentrum in Pixelkoordinaten (schwarzer Kreis mit Kreuz) ist markiert. Durch die geringere Anzahl der Sample-Points im Vergleich zum voll-aufgelöstem Bild, entsteht ein Fehler in der Zuordnung eines Sample-Point zu einer LED-Position. Somit entsteht eine Ungenauigkeit von  $\frac{15-1}{2} \pm 7$  Pixel für jede LED-Position in beide Richtungen. Beispielsweise wäre die größte Entfernung und somit die höchste Abweichung  $\sqrt{(\pm 7)^2 + (\pm 7)^2} < 9,8995$  Pixel. Es wird also jeder LED-Position  $pos(l)$  eine SP-basierte Position  $pos_{SP}(l)$  zugeordnet, welche der Position mit der kürzesten Entfernung zur tatsächlichen Position entspricht. An dieser Stelle sei nochmal angemerkt, dass die Modelle, ebenfalls in dieser Sample-Point Auflösung gespeichert werden sollen und folglich von dieser inhärenten Ungenauigkeit betroffen sind. Die Berechnung der SP-Position der LEDs mit Index  $l$  kann wie

Formel (5.9) gezeigt durchgeführt werden. Zunächst sei die Rundungsfunktion definiert.

$$\begin{aligned} \text{round} &: (\mathbb{R}_0^+)^n \rightarrow (\mathbb{N}_0)^n \\ \text{round}(\mathbf{x}) &\mapsto \begin{cases} \lfloor x_i \rfloor, & x_i < \frac{\lfloor x_i \rfloor + \lceil x_i \rceil}{2} \\ \lceil x_i \rceil, & \text{sonst} \end{cases}, \forall i \in \{0, \dots, n-1\} \end{aligned} \quad (5.8)$$

Diese Funktion wird also komponentenweise angewandt. Für  $n = 1$  ist das Resultat ein Skalar. Es ergibt sich damit die Komposition

$$\begin{aligned} \text{pos}_{\text{SP}} &: \{1, \dots, N_L\} \rightarrow \{1, \dots, N_{\text{SP}}^r\} \times \{1, \dots, N_{\text{SP}}^c\} \\ \text{pos}_{\text{SP}}(l) &= \text{round} \left( \text{pos}(l)^T \cdot \begin{pmatrix} \frac{N_{\text{SP}}^r}{N_{\text{P}}^r} & 0 \\ 0 & \frac{N_{\text{SP}}^c}{N_{\text{P}}^c} \end{pmatrix} \right), \forall l \in \{1, \dots, N_L\} \end{aligned} \quad (5.9)$$

Die Ergebnisse der Positionsberechnungen werden später (siehe Abschnitt 5.6) in einer LUT realisiert und im Hardware-Design verwendet.

### Sample-Point Berechnung der Modelle

Wie schon erläutert, müssen auch die Modelle auf die Auflösung der Stützstellen reduziert werden. Zunächst müssen dazu die SP-Positionen der LEDs anhand Formel (5.9) bestimmt werden. In Abbildung 5.27 sind zwei Methoden als Blockdiagramme dargestellt, um die Modelle auf Stützstellen zu reduzieren. Dabei sind äquivalenten Blöcke in gleicher Farbe eingefärbt. Methode 1 setzt wie in Abbildung 5.21 dargestellt an und schließt die Normierung der nativ aufgelösten Modelle mit der Abtastung an den Stützstellen ab. Daraus ergeben sich die Modelle auf SP-Basis ( $M_i^{\text{SP}'}$ ). Das Rechteck (orange) markiert hier den Bereich, indem statt auf der nativen Auflösung, mit Stützstellen gearbeitet wird. Die Methode 2 hingegen erzeugt direkt

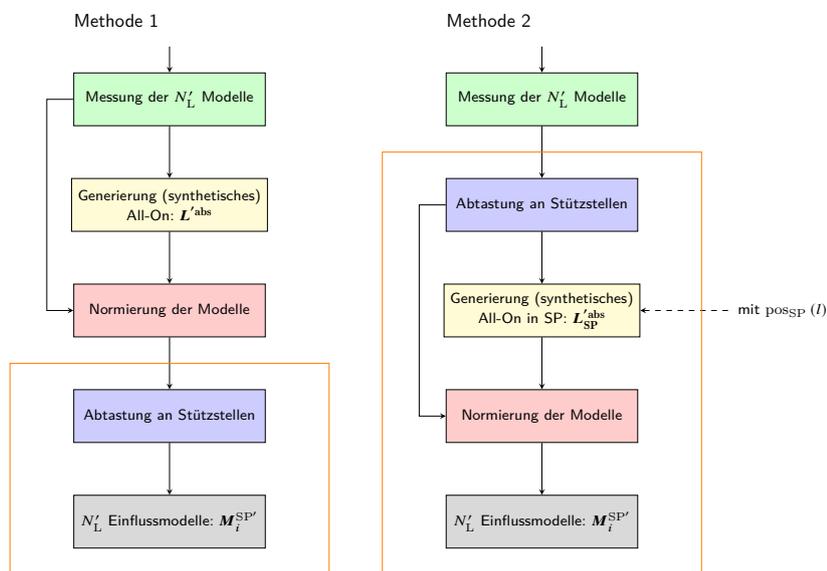


Abbildung 5.27.: Blockdiagramm der beiden Methoden zur Erzeugung der Sample-Points der Modelle.

aus den  $N'_L$  Messungen durch Abtastung eine SP-basierte Version dieser Messungen. Mithilfe der bekannten

SP-Positionen  $\text{pos}_{\text{SP}}(l)$  aller LEDs mit dem Index  $l$  wird direkt aus den auf SPs reduzierten Modellen ein synthetisches All-On erzeugt ( $L^{\text{abs}}$ ), welches analog zu Formel (5.7) zur Normierung genutzt wird die Einflussmodelle in SP-Auflösung liefert. Der Vorteil der letzten Methode ist, dass das All-On direkt in der korrekten SP-Auflösung berechnet wird und somit das Vorgehen in sich konsistent wird.

Der aus den finalen Modellen resultierende Einfluss einer LED  $l$  in allen Sample-Points wird als  $\check{L}_l^{\text{SP}'}$  bezeichnet. In folgenden Kapiteln wird zur Vereinfachung die Notation  $\check{L}_l'$  verwendet, wenn aus dem Kontext klar hervorgeht, ob es sich auf die Stützstellen bezieht oder auf die native Pixelauflösung. Analog hierzu wird auch  $M_i^{\text{SP}'}$  gehandhabt.

Wird anhand der 133er BLU der RMSE zwischen  $\sum_{l=1}^{N_L} \check{L}_l^{\text{SP}'}$  und  $\mathbb{1}^{N_{\text{SP}}^r \times N_{\text{SP}}^c}$  berechnet, so ergibt sich für die Methode 1 ein Wert von 0,0496 und für die Methode 2 einen Wert von 0,0359 und somit einen um  $\approx 27,65\%$  geringeren Fehler. Ein Beispiel des resultierenden SP-basierenden Modell 9 ist in Abbildung 5.28 abgebildet.

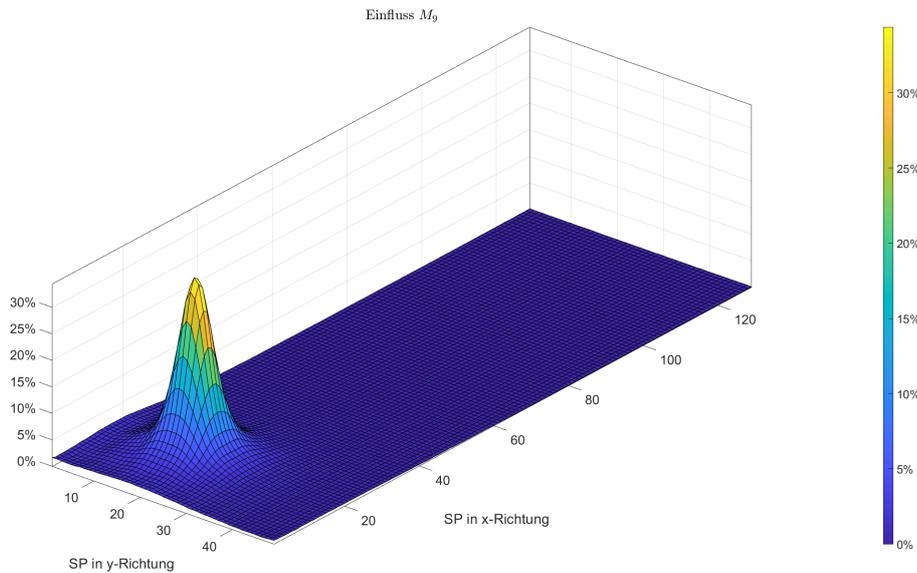


Abbildung 5.28.: Stützstellen-basierendes Einflussmodell  $M_0$ .

Bis hier wurde ein Modell an jedem Sample-Point gespeichert und stellt schon eine nennenswerte Reduktion des Speichers im Vergleich zur Speicherung in der nativen Pixelauflösung. Es wurde jedoch noch nicht berücksichtigt, dass der LSF Abfall (negative Steigung) mit zunehmender Entfernung flacher wird. Somit bringt eine hohe Sample-Point Dichte keine zusätzliche Information. Dieser Umstand kann ausgenutzt werden, um den Speicherbedarf weiter zu reduzieren, wie im nächsten Abschnitt demonstriert wird.

### 5.3.4. Verringerung des Modellspeichers durch reduzierte Stützstellendichte

Anhand Abbildung 5.28 ist gut erkennbar, dass der Einfluss des Modells ab einer bestimmten Entfernung sehr global mit wenig Änderung sowie – je nach LSF – auch sehr gering ist. Diese Feststellung legt nahe, die Anzahl der Stützstellen weiter zu reduzieren und den Speicherbedarf für die Hardwarenutzung der Modelle weiter zu verringern. Eine sinnvolle Grenze lässt sich anhand der *kurzen* Displayseite bestimmen und mithilfe der Sample-Point Nummerierung beschreiben. Handelt es sich, wie für ein ICD üblich, um ein Display im Querformat, so gilt  $N_{\text{SP}}^c > N_{\text{SP}}^r$ . Für ein Display im Hochformat gilt entsprechend  $N_{\text{SP}}^c < N_{\text{SP}}^r$ . Somit wird



sind und somit das Modell keine so genaue räumliche Auflösung benötigt.

Es ergibt sich also ein Speicherbedarf ( $\text{Speicher}_{\text{Modell}}$ ) bei einer Wertquantisierung von 12 Bit durch die Reduktion der räumlichen SP-Auflösung von

$$\begin{aligned}
 \text{Speicher}_{\text{Modell}} &= 12 \text{ Bit} \cdot \left( \underbrace{N_{\text{SP}}^r \cdot N_{\text{SP}}^r}_{49 \cdot 49 = 2401} + \underbrace{\left\lceil \frac{N_{\text{SP}}^r}{2} \right\rceil}_{25} \cdot \underbrace{\left\lceil \frac{N_{\text{SP}}^c - N_{\text{SP}}^r}{2} \right\rceil}_{40} \right) \\
 &\hat{=} 12 \text{ Bit} \cdot \left( \underbrace{2401}_{\text{SP-Auflösung}} + \underbrace{1000}_{\text{reduzierte SP-Auflösung}} \right) \\
 &= 40,812 \text{ kBit}
 \end{aligned} \tag{5.11}$$

Dabei sind unter den geschweiften Klammern die real genutzten Werte aus dieser Arbeit gezeigt. Für die 9 Modelle wird also unter Berücksichtigung der Reduktion der Stützstellendichte ein Speicher von  $40,812 \text{ kBit} \cdot 9 = 367,308 \text{ kBit}$  benötigt. Somit lässt sich der Modellspeicher um weitere 46 % reduzieren und entspricht damit nur noch 2,46 ‰ eines nicht realisierbaren Modellspeichers basierend auf der nativer (Pixel-) Auflösung des Panels.

Neben diesen neuen vorgestellten Methoden zur Speicherreduktion ergeben sich jedoch auch Implikationen auf die *Pixelorder* wie im nächsten Abschnitt gezeigt wird.

## 5.4. Scanning Sequenz und Struktur für eine neue Pixelorder

In Abschnitt 3.5.1 wurde die Funktionsweise des SSC-Algorithmus beschrieben. Es wurde deutlich, dass die *Pixelorder* einen sehr wichtigen Teil der Implementierung darstellt, da in dieser Datenstruktur die Abarbeitungsreihenfolge der einzelnen Pixel bzw. auf die Optimierung bezogen, der Condenserezellen gespeichert ist (siehe Abschnitt 3.5.1.2). Zusätzlich werden jeder Condenserezelle die LED-Nummern inklusive ihres Einflusses auf diese Zelle in absteigender Reihenfolge vom dominanten Einfluss gespeichert. Die einzelnen Einträge sind wie aus Abschnitt 3.5.1.2 bekannt, sortiert.

Aufgrund des SoA-Verfahrens mit  $N_C = N_{\text{SP}}$  steigt der Speicherbedarf der *Pixelorder* bei einer steigenden LED-Anzahl sowie steigender Condenserauflösung stark an. Der Bedarf wurde in Abbildung 5.6 auf Seite 134 bereits visualisiert. Aufgrund des neuen, modellbasierten Verfahrens und des starken Anstieges der *Pixelorder*-größe, wird die *Pixelorder* weiterentwickelt, um den Herausforderungen gerecht zu werden. In Abbildung 5.30 ist ein Abhängigkeitsdiagramm dargestellt, welches die Notwendigkeit einer neuen *Pixelorder* begründet. Ein höheres  $N_L$  führt zu zeitlichen Limitierungen und deutlich erhöhtem Speicherbedarf. Zudem wird aufgrund der steileren LSFs eine höhere SP-Auflösung benötigt, welches sich wiederum negativ auf die Berechnungszeiten und Speicherbedarf auswirken. Um das zeitliche Limit einhalten zu können, werden weniger LEDs pro Condenserezelle während der LED-Wert-Berechnung betrachtet, als in der BLU vorhanden sind ( $\tilde{N}_L \ll N_L$ ) wie in Abschnitt 5.2.2 vorgeschlagen wird. Zusätzlich erlaubt eine Trennung der in SoA identischen Condenserezellenanzahl  $N_C$  von der SP-Anzahl  $N_{\text{SP}}$  das zeitliche Limit einhalten zu können und reduziert zugleich den Speicherbedarf. Durch die Einführung von LSF-Modellen (siehe Abschnitt 5.3.1) wird der Speichergröße weiter verringert. Die genannten Methoden mit ihren Gegebenheiten führen jedoch

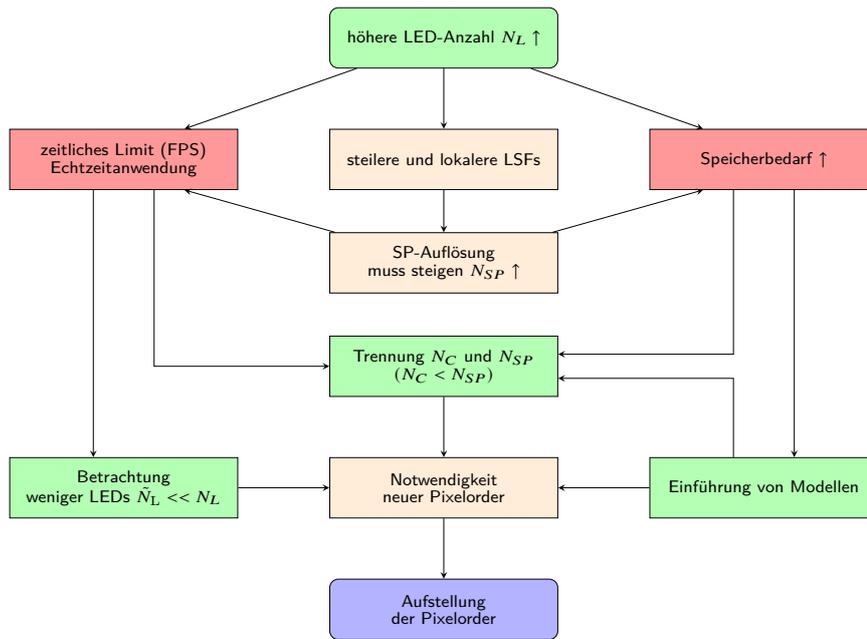


Abbildung 5.30.: Abhängigkeitsdiagramm und Implikation auf die Pixelorder.

zur Notwendigkeit der Neuentwicklung der Pixelorder Struktur und Abarbeitungsreihenfolge (*Scanning Sequenz*).

### 5.4.1. Strukturelle und inhaltliche Erweiterung für den Hardware-Einsatz

Ein wichtiges Ziel des neuen Lösungsansatzes basierend auf LSF-Modellen ist die Reduktion der Hardwarekosten pro LED bzw. pro LED-Strang für hochauflösende BLUs sowie die Einhaltung von zeitlichen Einschränkungen im Sinne von der Begrenzung durch die FPS. In der SoA-Implementierung kommt der Hardwarebedarf und die Berechnungszeit jedoch schnell an die Grenze der Umsetzbarkeit. In Abschnitt 5.1.2 *Grenzen der Hardware-Implementierung und resultierende Herausforderungen* wurde darauf im Detail eingegangen. Unter Annahme aus Formel (5.2) wird für die effiziente Hardware-Implementierung die Anzahl der berücksichtigten LEDs zu  $\tilde{N}_L = 25$  gesetzt, welches einen guten Kompromiss zwischen Effizienz und Powersaving bei adäquater Iterationszahl liefert. In Abschnitt 5.2.2 wurden die Auswirkungen der Wahl von  $\tilde{N}_L$  analysiert.

Somit muss pro Pixelorderzeile bzw. Pixelorderadresse mindestens die LED-Nummer mit dem höchsten Einfluss auf die aktuell betrachtete Condensierzelle in der Pixelorder gespeichert werden. Diese Haupt-LED wird als  $LED_{p,l}^M$  beschrieben. Analog zu Formel (4.3) entspricht  $p$  dem linearen Index der betrachteten Condensierzelle statt eines einzelnen Pixels. Für eine Direct-Lit BLU lässt sich diese LED-Nummer durch den Abstand zur betrachteten Condensierzelle bestimmen, da in diesem Fall der höchste Beitrag *über* dem Zentrum einer LED liegt. Anhand Abbildung 5.26 erkennt man, dass das Zentrum einer Condensierzelle immer auf einen Sample-Point fällt. Somit ergibt sich die Condenserkoordinate in SP-Koordinaten bzw.

Auflösung ausgedrückt zu

$$C_{r,c}^{\text{SP}} = C_{2 \cdot i, 2 \cdot j} \\ \Rightarrow r = i \ll 1 \wedge c = j \ll 1 \quad (5.12)$$

$$\text{mit } (r, c) \in \{1, \dots, N_{\text{SP}}^r\} \times \{1, \dots, N_{\text{SP}}^c\} \wedge (i, j) \in \{1, \dots, N_C^r\} \times \{1, \dots, N_C^c\}$$

und entspricht einem binären *Shift* ( $\ll$ ) der  $(i, j)$  Koordinaten in der Condenserauflösung um 1 Bit nach links. Zusätzlich kann eine Condenserezelle auch linear adressiert werden. Zusammen mit der SP-Position  $\text{pos}_{\text{SP}}(l)$  einer LED  $l$  (siehe Formel (5.9)) wird die Haupt-LED für alle Condenserezelle mit linearem Index  $p$  gefunden

$$\forall C_p \exists! \text{LED}_{p,l}^M : \|\text{pos}_{\text{SP}}(\text{LED}_l) - \text{pos}_{\text{SP}}(C_p)\| < \|\text{pos}_{\text{SP}}(\text{LED}_k) - \text{pos}_{\text{SP}}(C_p)\|, \forall k \neq l \quad (5.13)$$

Dabei ist  $k, l \in \{1, \dots, N_L\}$ . Da für in der Automobilanwendung die Sicherheit und Bildqualität eine sehr wichtige Rolle spielen, wird in der Implementierung in Abschnitt 5.6 nur der bereits vorgestellte Maximalwertcondenser genutzt, um eine konservative Lösung des für die berechneten LED-Werte zu erhalten. Um jedoch das Powersaving zu erhöhen, wird ein Parameter eingeführt, welcher die Anzahl derjeniger LEDs reguliert, welche zum Ausgleich des fehlenden Lichtbeitrags einer Condenserezelle  $C_p$  verwendet werden dürfen. Der Wahlbereich wurde dabei zwischen 1 und 5 festgelegt. Diese werden mit  $\text{LED}_{p,l}^M$  bis  $\text{LED}_{p,k}^{M-4}$  bezeichnet. Dabei bedeutet  $\text{LED}_{p,l}^M$ , dass die LED  $l$  auf die Condenserezelle oder Pixel  $p$  den größten Einfluss hat (Main-LED).  $M - 1$  steht für den zweitgrößten,  $M - 2$  für den drittgrößten Beitrag usw.

Für das Hardwaredesign hat sich für den Berechnungsablauf als sinnvoll herausgestellt, dass die 4 weiteren LED-Nummern zusätzlich zu der Haupt-LED aus der Menge  $\mathcal{M}_{i,j}$  der betrachteten LEDs für eine Condenserezelle, gespeichert werden (linearer Index  $p$  entspricht hier den Koordinaten  $(i, j)$ ). Die zusätzlichen LEDs entsprechen den beiden horizontalen sowie vertikalen Nachbarn der Haupt-LED. Diese feste Zuordnung erlaubt eine systematische und effiziente Berechnung. Zudem müssen die LED-Nummern nach maximalen Einfluss sortiert werden, um zu garantieren, dass sobald die Haupt-LED den Maximalwert erreicht hat, die nächstmögliche optimale LED erhöht werden kann. Somit wird eine Reihenfolge der Einträge in der Pixelorder bestimmt, welche dem Algorithmus bekannt ist. Es gilt für die Einflüsse dieser 5 LEDs

$$\underbrace{\check{L}_{p,k}}_{\text{LED}_{p,k}^M} \geq \underbrace{\check{L}_{p,l}}_{\text{LED}_{p,l}^{M-1}} \geq \underbrace{\check{L}_{p,m}}_{\text{LED}_{p,m}^{M-2}} \geq \underbrace{\check{L}_{p,n}}_{\text{LED}_{p,n}^{M-3}} \geq \underbrace{\check{L}_{p,o}}_{\text{LED}_{p,o}^{M-4}} \quad \text{mit } k \neq l \neq m \neq n \neq o \quad (5.14)$$

Die Menge dieser 5 LEDs wird als  $\mathcal{M}_{i,j}^{\text{inc}}$  bezeichnet. In Abbildung 5.31 wird anhand eines Beispiels visualisiert, wie die LED Nummern zu einer Condenserezelle zugeordnet werden. Diese werden dann jeweils in einer Pixelorderzeile gespeichert. Die weiteren LEDs aus  $\mathcal{M}_{i,j} \setminus \mathcal{M}_{i,j}^{\text{inc}}$  bzw. deren LED-Nummern werden in der Abarbeitungspipeline der Pixelorder im später vorgestellten *Optimizer* berechnet. In der genannten Abbildung ist eine beispielhafte Situation dargestellt. Grün markiert sei eine Condenserezelle mit Index  $p$ . In Rot ist die Haupt-LED dargestellt sowie in Orange die weiteren 4 LEDs, welche bei Bedarf erhöht werden dürfen. Diese insgesamt 5 LEDs sind diejenigen, die in der Pixelorder gespeichert werden. Die übrigen, in Schwarz dargestellte LEDs bzw. deren Nummern, werden zu Laufzeit der Optimierers bei jedem Einlesen einer Pixelorderzeile nach Geometrie berechnet. Da nur maximal 5 LEDs pro Zelle angepasst werden dürfen, ist es auch sinnvoll nur diese 5, anstelle der 25 betrachteten LEDs an dieser Stelle zu speichern. Das eingeführte Verfahren der LSF Modelle verhindert unter anderem diese Abhängigkeit der Pixelorder von der

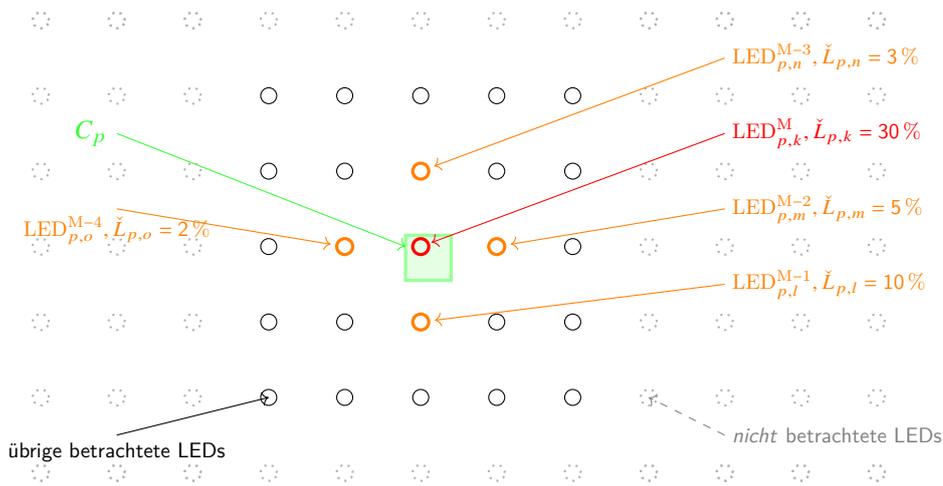


Abbildung 5.31.: Beispiel der Zuordnung der LEDs zu einer Condensierzelle  $C_p$  mit beispielhaften Werten.

LED-Anzahl  $N_L$ , da die Einflüsse aller betrachteten LEDs durch eine später eingeführte Adressierlogik aus diesen Modellen entnommen werden können. Diese Einflüsse dienen dazu, den noch fehlenden Lichtbeitrag zu einer Condensierzelle  $C$  mit Index  $(i, j)$  und Wert  $c(i, j)$  zu berechnen wie in Formel (5.15) beschrieben. Da die Definitions- und Wertebereiche bekannt sind, wird auf die explizite Angabe verzichtet. Die Funktion  $\lambda(\mathbf{c}, t)$  mit  $t \in \{1, \dots, N_{\text{iter}}\}$  erhöht in jedem Iterationsschritt  $t$  sukzessive den Condenservwert (Zielwert für den Soll-Lichtbeitrag), sodass im letzten Iterationsschritt gilt  $\lambda(\mathbf{c}, N_{\text{iter}}) \stackrel{!}{=} \mathbf{c}$ . Dadurch erfolgt wie in Abschnitt 3.5.1.2 beschrieben eine sukzessive Annäherung der Lösung an das Optimum, da der Crosstalk effektiv berücksichtigt wird.

$$\underbrace{\Delta_{\check{L}}(i, j)}_{\text{Differenz zum Soll-Wert}} = \underbrace{\lambda(\mathbf{c}(i, j), t)}_{\text{Soll-Wert}} - \underbrace{\sum_{m \in M_{i,j}} \check{L}_{(i,j),m} \cdot \mathbf{v}(m)}_{\text{aktueller Lichtbeitrag/virtuelles Backlight}} \quad (5.15)$$

Insgesamt ist ein Pixelorderwort also 112 Bit breit. Die Anzahl der Adressen bzw. Einträge ist  $N_C$ . In diesem Abschnitt wurde die Pixelorder und insbesondere der Inhalt und Sortierung innerhalb einer Pixelorderzeile vorgestellt und die Notwendigkeit einer Anpassung aufgrund der Direct-Lit Herausforderungen erläutert. Im nächsten Abschnitt wird auf den Einfluss Sortierung der Zeilen untereinander eingegangen.

### 5.4.2. Selektion der Scanning Sequenz

Aufgrund des iterativen Charakters zur Lösung des grundlegenden Optimierungsproblems (Formel (5.3)) in einer Echtzeitanwendung wird die Wirkung von verschiedenen Iterationszahlen sowie die Wahl der Abarbeitungsreihenfolge der Pixelorder (*Sortierung*) untersucht. Eine Sortier- und Abarbeitungsmethode ist in Abbildung 5.32 nach dem Vorschlag von Albrecht [Alb10] dargestellt. Die Zellen werden nach dem Wert des LSF-Einflusses in dieser Zelle sortiert. Die Zelle 1 ist in der Gruppe der LED 1 und wird als erster betrachtet. Im nächsten Schritt wird eine Zelle aus dem Bereich der LED 2 mit dem maximalen Einfluss betrachtet usw. bis alle Zellen der Sektoren mit den höchsten Einflüssen abgearbeitet sind. Danach ist die Zelle mit dem zweithöchsten Einfluss der LED 1 an der Reihe usw. Wurden alle Condensierzellen abgearbeitet, entspricht diese einer Iteration über die Pixelorder. Dieser Durchlauf wird mehrfach wiederholt unter Anwendung

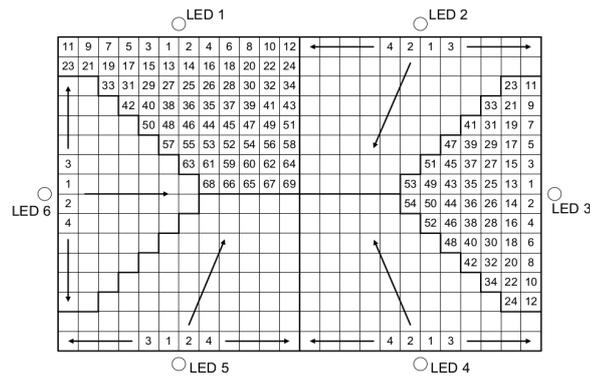


Abbildung 5.32.: Abarbeitungsreihenfolge der Condensierzellen für ein Edge-Lit Gerät. Entnommen aus [Alb10].

der Funktion  $\lambda(c, t)$  mit  $t \in \{1, \dots, N_{\text{iter}}\}$  (siehe Formel (5.15)). In [Sch17] wurde die Sortierung noch weiter optimiert und die Beschränkung der Reihenfolge etwas gelockert, sodass jetzt nicht nur in den ersten  $N_L$  Einträgen der Pixelorder alle LEDs genau einmal die LEDs mit höchsten Einflüssen in einer Zelle waren, sondern auch zusätzlich nach ihrem maximalen Einfluss abgearbeitet werden. Es musste also nicht mehr zwangsläufig die Reihenfolge der LED1  $\rightarrow$  LED2  $\rightarrow$  LED3 ... sein, die die Condensierzellenreihenfolge bestimmt.

Für diese Berechnungen wird hier exemplarisch eine 319er BLU verwendet. Aufgrund der hohen LED-Anzahl ergeben sich viele möglichen Scanning-Sequenzen. Zur Visualisierung ist in Abbildung 5.33 die Zuordnung einer LED-Nummer zu jeder Condensierzelle dargestellt (vgl. Abbildung 5.31). Die resultierenden breiteren *Streifen* werden durch die Positionierung der LED-Zentren auf SP-Basis wie in Abschnitt 5.3.3 eingeführt, hervorgerufen.

Zum Vergleich von verschiedenen Sortier- bzw. Abarbeitungsreihenfolgen der Pixelorder, werden 6 verschiedene Sortierungen erzeugt. Diese werden durch die Zuordnung der Pixelorderadresse  $\mathbf{addr}(i), \forall i \in \{0, \dots, N_C - 1\}$  zu der entsprechenden Condensierzelle  $C_p$  mit  $p \in \{0, \dots, N_C\}$  beschrieben. Der Index steht repräsentativ für eine Condensierzelle. Die Sortierungen sind wie folgt gewählt und  $\exists \mathbf{addr} \dots$  soll

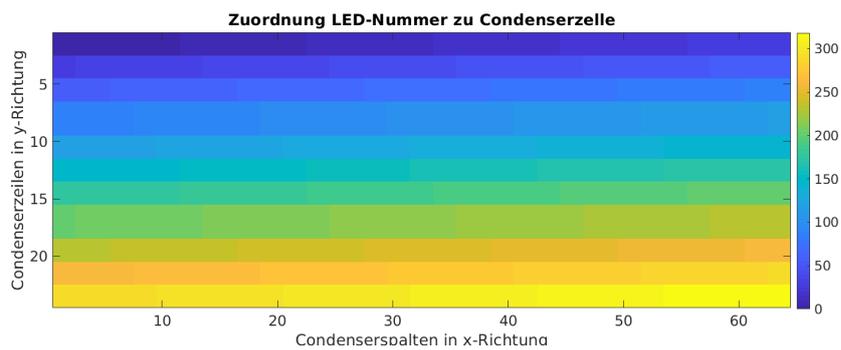


Abbildung 5.33.: Zuordnung der Haupt-LEDs zu den Condensierzellen.

interpretiert werden als: *Es gibt einen Vektor  $\mathbf{addr}$  (welcher an der Position  $i$  den zugehörigen Zellenindex beinhaltet), sodass gilt ...*

1. Sortierung: zufällig
2. Sortierung: zufällig
3. Sortierung: Zellennummern linear aufsteigend  
 $\forall i : \mathbf{addr}(i) = i$
4. Sortierung: nach Haupteinfluss und LED-Verteilung nach Schäfer [Sch17]  
 $\exists \mathbf{addr} \forall n \in \{0, \dots, \lfloor \frac{N_C}{N_L} \rfloor\} \wedge i \in \{n \cdot N_L, \dots, (n+1) \cdot N_L\} :$   
 $\{0, \dots, N_L\} \stackrel{!}{=} \{\mathbf{addr}(n \cdot N_L), \dots, \mathbf{addr}((n+1) \cdot N_L)\} \wedge \check{L}_{i,k} \geq \check{L}_{i+1,l}$   
 mit  $k \neq l \in \{0, \dots, N_L - 1\}$   
 Damit wird ein Constraint eingeführt, welcher dafür sorgt, dass eine LED erst wieder betrachtet werden darf, wenn alle Haupt-LEDs bereits betrachtet wurden.
5. Sortierung: nach Haupteinfluss  
 $\exists \mathbf{addr} \forall i : \check{L}_{i,k} \geq \check{L}_{i+1,l}$  mit  $k, l \in \{0, \dots, N_L - 1\}$
6. Sortierung: Zellenabstand von der Display Mitte (Zelle mit Index  $z$ )  
 $\exists \mathbf{addr} \forall i : \|C_i - C_z\| \leq \|C_{i+1} - C_z\|$

In Abbildung 5.34 sind diese 6 Sortierungen durch die Abarbeitungsreihenfolge dargestellt. Hier wird jeder Condenserszelle, repräsentiert durch ihre Zeilen- und Spaltenlage, die Adresse in der Pixelorder, also ihre Abarbeitungsnummer zugeordnet. Auffällig ist bei der 5. Sortierung, dass zunächst die Condenserszellen am

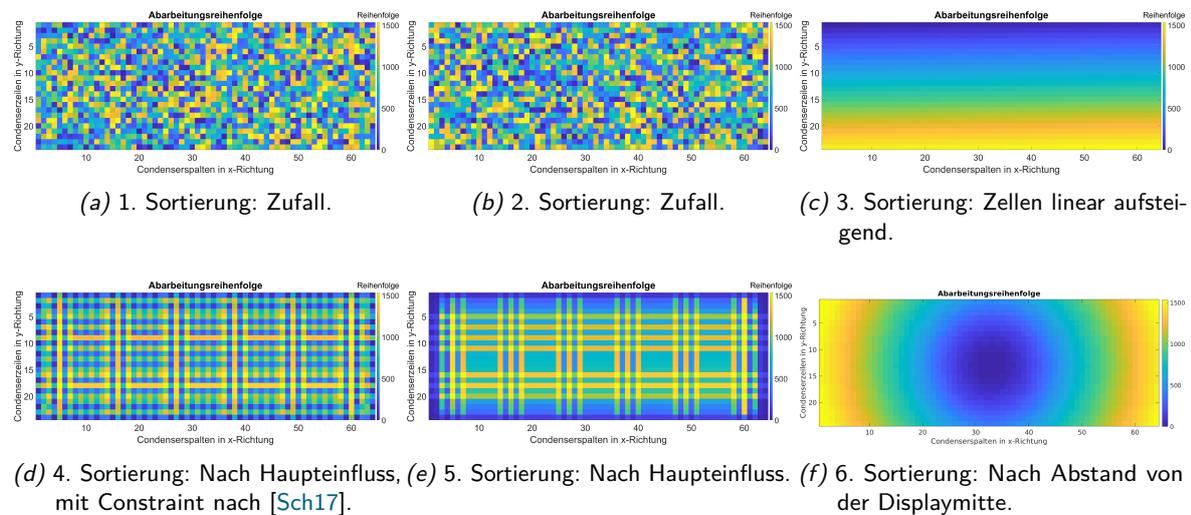


Abbildung 5.34.: Verschiedene Abarbeitungsreihenfolgen der Pixelorder.

Rand betrachtet werden. Die Ursache liegt hier daran, dass am Rand der maximale Einfluss der LEDs in den Condenserszellen am höchsten ist, da von *außerhalb* des Displays kein Licht eingekoppelt wird. Die Abbildung 4.11 auf Seite 83 sind die maximalen Einflüsse dargestellt und zeigt diesen Effekt auf.

Bei der 4. Sortierung ist die Wirkung des zusätzlichen Sortierconstraints deutlich erkennbar. Dazu werden in Abbildung 5.35 die Verteilungen der absoluten Häufigkeit der Haupt-LED Betrachtungen in den ersten  $N_L$  (Anzahl der LEDs) Zeilen der Pixelorder in Form von Histogrammen aufgezeigt. Insbesondere zeigt das Histogramm zur Pixelordersortierung 4 deutlich, dass jede LED genau einmal als Haupt-LED betrachtet

wird. Bei der linearen Abarbeitung hingegen, werden bis zu diesem Schritt nur wenige LEDs in der oberen Hälfte des Displays betrachtet.

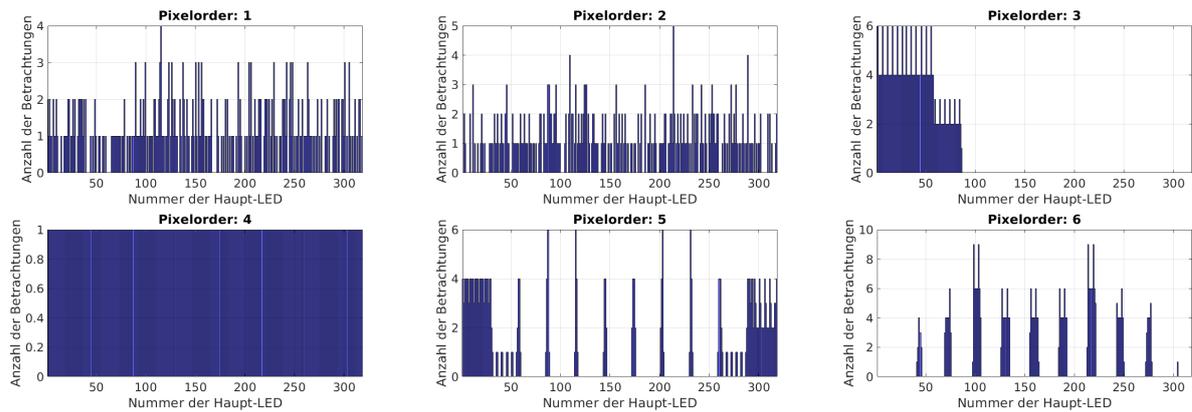


Abbildung 5.35.: Verteilungen der Haupt-LEDs in den ersten  $N_L$  Zeilen der Pixelorder (LED-Nummern linear indiziert).

In Abbildung 5.36 sind 6 typische HMI Inhalte als Testbilder ausgewählt. Zur Berechnung der LED-Werte wird nach Formeln (3.5) und (5.15) die oben erwähnte Funktion  $\lambda(c, t)$  so realisiert, dass die in der aktuellen Iteration  $t$  betrachteten Condenserwerte  $c'_t$  wie folgt berechnet werden

$$c'_t(i, j) = \lambda(c(i, j), t) = c(i, j) - s(t) \quad (5.16)$$

Die Zuordnung  $s(t)$  ordnet jeder Iteration eine zu subtrahierende Schwelle zu. Diese können verschieden gewählt werden. Sinnvoll im Sinne der iterativen Annäherung der LED-Wert Lösungen, dürfen die Anfangswerte  $c'_1(i, j)$  nicht zu groß sein, da ansonsten die entsprechenden LEDs unnötigerweise erhöht werden. Gerade im ersten Durchlauf sind die aktuellen Lichtbeiträge sehr gering und bei großem Zielwert werden die Haupt-LEDs entsprechend zu hoch berechnet, um den vermeintlich fehlenden Lichtbeitrag auszugleichen. Der Crosstalk der LEDs wird jedoch generell in jedem Schritt berücksichtigt und stellt ein besonderes Merkmal des ursprünglichen SSC-Algorithmus dar. In den folgenden Iterationen nähert sich  $c'_t(i, j)$  den tatsächlichen Werten. Es gilt also

$$\lim_{t \rightarrow N_{\text{iter}}} c'_t = c$$

Für diesen Versuch wird die Iterationszahl  $N_{\text{iter}}$  variiert und die Schwellen so gewählt, dass die Abstände  $s(t) - s(t+1)$ ,  $\forall t \in \{1, \dots, N_{\text{iter}} - 1\}$  äquidistant sind für die verschiedenen gewählten Iterationszahlen. Um die Berechnungszeit zu verkürzen, werden die Condenserwerte von 10 Bit auf 8 Bit gekürzt.

Aufgrund der Reduktion der Zielwerte auf 8 Bit ist eine maximale Iterationszahl von 255 gegeben, wenn gilt  $s(t) - s(t+1) = 1$  und entspricht jeweils einer Änderung der Schwelle um 1 Bit. Es wurden Iterationszahlen aus  $\{1, 3, 5, 10, 15, 20, 40, 80, 160, 255\}$  verwendet und die Schwellen entsprechend mit linearem Abstand angepasst. Daraus ergeben sich die folgenden Powersavingpotentiale in Abbildung 5.37 für die 6 oben gezeigten HMI-Bilder für jeweils alle aufgeführten Pixelorder Sortierungen. Insgesamt lässt sich die Tendenz erkennen, dass eine höhere Iterationszahl und somit eine feiner aufgelöste  $\lambda(c, t)$  insbesondere im niedrigen Iterationsbereich hohen Einfluss auf das Powersaving hat. Es wird auch deutlich, dass die Abarbeitungsreihenfolge für die meisten Bilder nicht zu vernachlässigen ist. Insbesondere wird dies bei

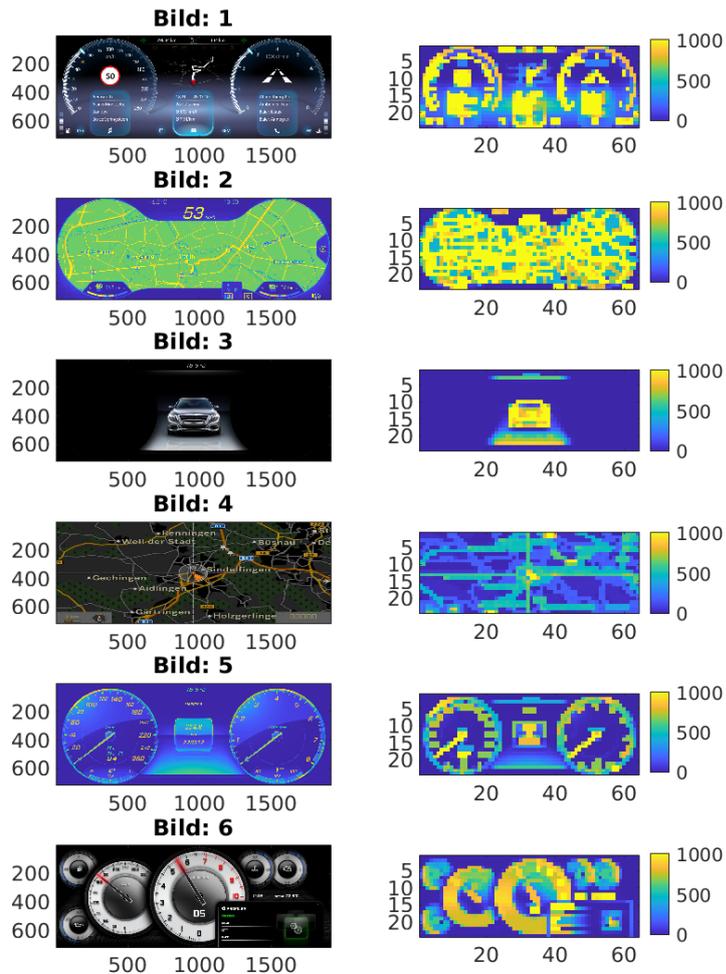


Abbildung 5.36.: Ausgewählte Testbilder (links) und resultierender Condenser-RAM-Inhalt (rechts) dargestellt in 10 Bit Auflösung.

genau einer Iteration deutlich, da hier direkt der volle Zielwert (Condenserwert) erreicht werden soll. Bei Bild 1, welches im Vergleich zu den übrigen kaum schwarze bzw. dunkle Flächen hat, liegt das beste und das schlechteste Ergebnis nur 1,74 Prozentpunkte auseinander. Bei Bild 4 beispielsweise beträgt der Unterschied 14,11 Prozentpunkte und entspricht einer Verbesserung von  $\approx 41\%$ . Für höher werdende Iterationszahlen gleichen sich die Resultate an. Aufgrund der bekannten zeitlichen Herausforderungen, wird in der Hardwareimplementierung jedoch eine Iterationszahl  $< 10$  verwendet, sodass die Sortierung, d.h. die Pixelorder relevant bleibt. Die Abbildung A.8 (Seite 246) zeigt als Beispiel die finalen LED Werte bei verschiedenen Iterationszahlen für das Bild 6.

Eine weitere Erkenntnis ist auch, dass die Auswirkung der Sortierung und der Iterationszahl stark von dem Bildinhalt abhängt. Bei Bild 2 und 3 lässt sich an der Skala erkennen, dass zwischen  $N_{\text{iter}} = 1$  und  $N_{\text{iter}} = 255$  der Unterschied im Ergebnis deutlich geringer ist, als bei den übrigen Testbildern. Somit hängen sie weniger von der Iterationszahl ab, als die übrigen und bedeutet, dass eine feiner aufgelöste  $\lambda(\mathbf{c}, t)$  bei diesen Bildern keine große Verbesserung bewirkt. Zudem wird deutlich, dass die lineare Abarbeitung der Condenserzellen sowie die einflussunabhängige Sortierung nach Abstand von der Displaymitte, zu den meisten Bildern ein unterdurchschnittliches Powersaving liefern. Auf der anderen Seite liefert die Sortierung

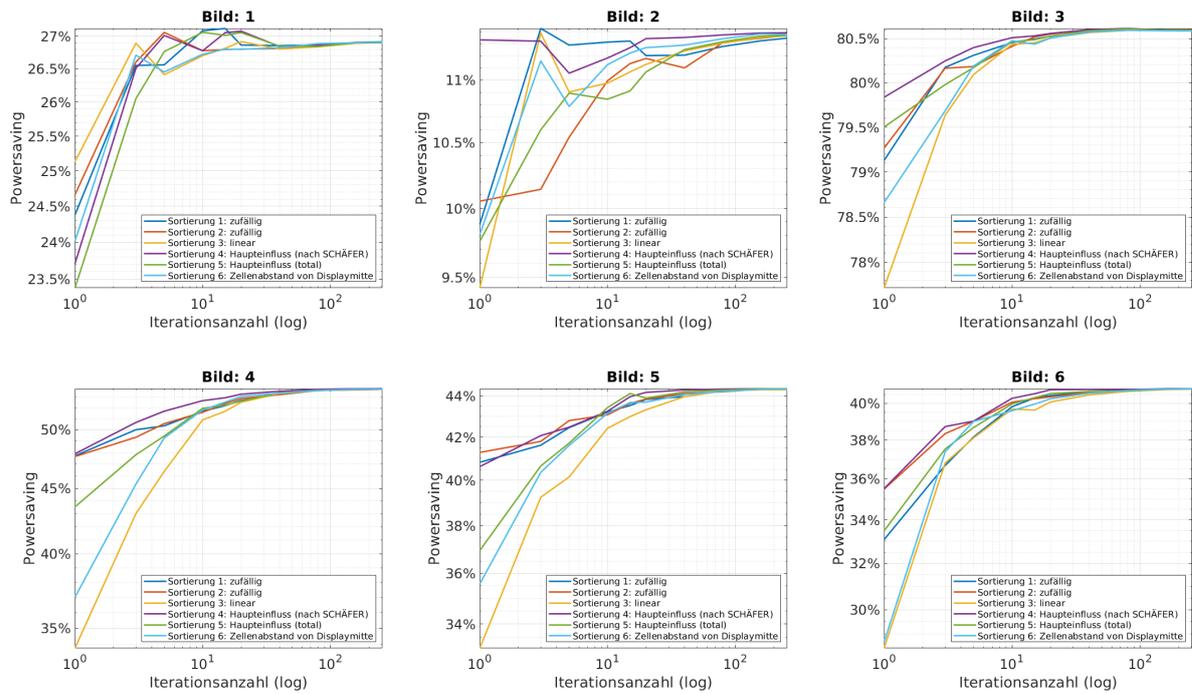


Abbildung 5.37.: Potential der Reduktion des Leistungsumsatzes der 6 ausgewählten Sortierungen zu den oben gezeigten Testbildern.

4 in den meisten Fällen die beste Performance. Auffällig ist bei Bild 2, dass schon bei einer Iterationszahl von 1, diese Abarbeitungsreihenfolge als einzige mit Abstand, einen um 20 % höheren Wert liefert als die lineare Sortierung. Dieses Verhalten erklärt sich dadurch, dass das Bild über die Fläche verteilt hohe Condenserwerte hat. Zunächst werden die Condenserezellen prozessiert, welche den maximalen Einfluss einer LED bekommen. Es wird jedoch garantiert, dass die Reihenfolge so gewählt ist, dass alle LEDs einmal als Haupt-LED einer Condenserezelle prozessiert sein müssen, bevor eine LED ein weiteres mal an der Reihe ist. Anders verhält sich die Sortierung 5 rein nach den Haupteinflüssen. Hier wird in Abbildung 5.34 deutlich, dass zunächst die LEDs am Rand prozessiert werden, dadurch werden sie von den hellen, am Rand liegenden Zellen zu hoch berechnet, da der tatsächliche Lichtbeitrag, welcher von den mittleren LEDs beigetragen wird, als 0 angenommen werden muss, da diese LEDs im Zentrum erst nach den Rändern prozessiert werden. Zum Vergleich der Performance wird das mittlere Powersaving der 6 Sortierungen mit 43 Testbildern (siehe Anhang B) berechnet. Als Benchmark dient der Durchschnitt der zufälligen Sortierungen was eine nicht systematische Reihenfolge ergibt. In Tabelle 5.5 sind die prozentualen Abweichungen von dem Benchmark dargestellt.

Benchmark	Abweichung gegenüber Benchmark			
Ø Sort. 1,2	Sort. 3	Sort. 4	Sort. 5	Sort. 6
<b>42,1 %</b>	-7,3 %	2,6 %	-2,5 %	-4,1 %

Tabelle 5.5.: Mittleres Powersaving und initiale Performance der Sortierung bei einer Iteration.

Nur die Sortierung 4 schafft es, die Zufallsreihenfolge (bei einer Iteration) zu übertrumpfen. Wohlgermerkt,

ist der Nutzen einer adäquaten Abarbeitungsreihenfolge vom Bildinhalt abhängig aber trotzdem sinnvoll, da der Algorithmus wie schon erwähnt, mit einer niedrigen Zahl von Iterationen auskommen muss und in diesem Bereich die Unterschiede der Sortierungen deutlich stärker ins Gewicht fallen.

Neben der Einführung einer an die neuen Methoden neu entwickelte Pixelorderstruktur, zeigte die Analyse der Scanning Sequenz, dass die Ergebnisse für große  $N_{iter}$  gegen das tatsächliche Optimum laufen. In Hardware ist die Iterationszahl jedoch aufgrund der zeitlichen Limitierung gering ( $N_{iter} < 10$ ). In diesem Bereich spielt die Scanning Sequenz, also die Abarbeitungsreihenfolge der Condensierzellen eine größere Rolle, je nach genutzter Sortierung. Zudem konnte gezeigt werden, dass eine *lineare* Sortierung (Sort. 3) mit Abstand die größte (negative) Abweichung von einer *zufälligen* Sortierung liefert und somit nicht genutzt werden sollte.

Somit wird ersichtlich, dass die vorgestellte Sortierung 4 aufgrund des Vorteils beim Powersaving gegenüber dem Benchmark, als auch wegen des Constraints der „fairen“ Betrachtung der LEDs genutzt werden soll.

Als Vorbereitung auf die Diskussion der neuen Local-Dimming Systemarchitektur (Abschnitt 5.6) wird im nächsten Abschnitt ein Feature des Systems vorgestellt, um flexibel auf Bedarf oder Wunsch von Zulieferern oder OEMs reagieren.

## 5.5. LED-Clustering

Aus Abschnitt 3.5.3 ist bekannt, dass bei einer hohen LED-Anzahl, insbesondere in der Single-Channel Ansteuerung, der Verdrahtungsaufwand hoch ist und Treiber mit sehr vielen Kanälen erforderlich werden. Dadurch steigen auch die Kosten. Aus diesem Grund wird eine Methode eingeführt, um diesen Aufwand reduzieren zu können. Diese Reduktion der Treiberkanäle wird dadurch erreicht, dass mehrere LEDs gemeinsam mit demselben Signal (meistens PWM) angesteuert werden. Dadurch wird nur noch ein gemeinsamer Treiberkanal benötigt und Verdrahtung vereinfacht sowie die Verlustleistung des Treibers reduziert. Diese Flexibilität des im Rahmen dieser Arbeit entwickelten Local-Dimming Algorithmus hat einen hohen Stellenwert und liefert einen Vorteil, da auf verschiedene BLUs/Treiber/Verdrahtungs-Konstellationen mithilfe von *Parametern* adaptiert werden kann, ohne das Hardwaredesign oder den IC ändern zu müssen. Diese *LED-Kopplung* wird hier auch als *LED-Clustering* bezeichnet. Ein wichtiges USP ist jedoch dabei, dass die einzelnen LEDs – trotz Kopplung – während der Optimierung einzeln betrachtet werden. Dadurch wird die Möglichkeit geschaffen, auch nicht direkt benachbarte LEDs zu einer virtuellen Einheit zu koppeln. Die Alternative nur die gekoppelten LEDs als ganze Einheit zu betrachten hat den großen Nachteil, dass die LSFs dieser gekoppelten Einheit gemessen werden müssten und dadurch sinnvollerweise nur die direkten Nachbarn gekoppelt werden könnten, um dem in dieser Arbeit vorgestellten Modellansatz zu genügen. Zudem wäre dadurch eine Verwendung in verschiedenen Displaymodellen umständlich, da je nach Kopplung die LSFs neu gemessen werden müssten. Ein weiterer Nachteil von der Betrachtung als eine Clustereinheit wäre die Notwendigkeit der Änderung in dem Modellansatz und auch eine Einschränkung in der Wahl des LED-Clusters. In Abbildung 5.38 ist eine Beispielkopplung der LEDs aus der 133er BLU dargestellt. Bis auf die LEDs 9, 27, 46, 65, 84, 109 sowie 123, die einzeln angesteuert werden, hat in diesem Beispiel jede weitere LED einen horizontalen Kopplungspartner. Die hellgrün dargestellten LEDs immer mit ihrem direkten Nachbarn gekoppelt. Das Besondere an dieser Auswahl ist die Kopplung der oberen bzw. unteren Eck-LEDs miteinander. Diese flexible Auswahl wird durch die beschriebene Betrachtung der *einzelnen* LEDs während

der Prozessierung ermöglicht. Der hieraus resultierende Vorteil ist die Möglichkeit die Clusterung an den geplanten Bildinhalt zu adaptieren. Bei den typischen automotive HMIs welche auf einem ICD dargestellt werden, befinden sich in den Ecken meistens keine Information, da hier der Bereich zur besseren Integration des Displays ins Innendesign, meist dunkel gewählt wird. Für die Hardwareimplementierung muss dieser

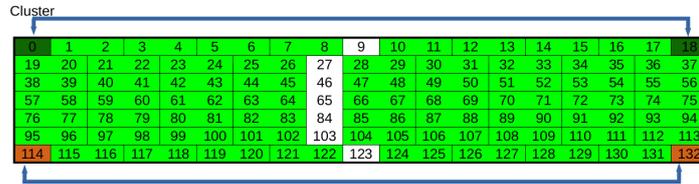


Abbildung 5.38.: Beispiel eines möglichen Clusterings der 133er BLU.

Information dem Algorithmus gekannt sein. Folglich wird eine Cluster-Mapping-Funktion benötigt, welche jeder LED-Nummer eine Koppel-LED ( $LED_k$ ) zuordnet. Diese ist hier dargestellt:

$$\text{map}_{LED}^{LED_k} : \{1, \dots, N_L\} \rightarrow \{1, \dots, N_L\}$$

$$l \mapsto k$$

Sollen keine LEDs gekoppelt werden, handelt es sich um die Identität. Somit wird jeder LED-Nummer ihre eigene Nummer zugeordnet ( $l \mapsto l$ ), sodass sich die Prozessierungsschritte nicht ändern und das Hardwaredesign folglich nicht angepasst werden muss. Für die Realisierung in Hardware bietet sich an, die vorhandenen Datenstrukturen bzw. ROMs zu erweitern. In dieser IP wird die vorhandene Pixelorder um die Clustering Information erweitert. Diese wird als zusätzliche LED-Nummer gespeichert, um die obige Mapping-Funktion zu realisieren. Jeder Haupt-LED wird also eine Kopplungs-LED zugeordnet. Des Weiteren wird der gespeicherte reziproke Einfluss jetzt gespeichert als der Kehrwert der summierten Einflüsse der gekoppelten LEDs. Ist  $l$  ein LED-Index und  $k$  der Index der gekoppelten LED so wird der Kehrwert der Einflüsse in Condensierzelle  $p$  gespeichert als  $1/(\check{L}'_{p,l} + \check{L}'_{p,k})$ .

Somit erlaubt diese Local-Dimming Technologie eine einzige BLU mit verschiedenen Treiberkonstellationen zu betreiben, ohne dass eine neue Messung benötigt wird. Auch ein neues Hardwaredesign entfällt, da der Algorithmus die BLUs mit und ohne Clustering identisch behandelt und bis auf die Cluster-Map mit derselben Konfiguration betrieben werden kann. So gibt es auch die Freiheit die Cluster-LEDs beliebig zu wählen, ohne auf den direkten Nachbarn eingeschränkt zu sein. Es muss jedoch sichergestellt werden, dass nachdem eine Haupt-LED erhöht wurde, die entsprechende gekoppelte LED auf denselben Wert gesetzt wird, damit der resultierende Crosstalk im nächsten Schritt Berücksichtigung findet.

In Abbildung 5.39 ist zum Clustering-Konzept ein Beispiel abgebildet. Gezeigt wird ein Testbild in Falschfarben sowie die resultierenden Lichtverteilungen basierend auf einer 96-LED BLU. Dabei wird ersichtlich, dass die Clustering Methode der direkten horizontalen Nachbarn (unten rechts) das schlechteste Ergebnis liefert. Hier muss die obere linke LED denselben Wert besitzen, wie der direkte Nachbar. Somit wird der eigentlich dunkle Ecke trotzdem hell ausgeleuchtet. Für die übrigen Ecken gilt es analog. Das Powersaving ist in diesem Fall ebenfalls um knappe 10 Prozentpunkte schlechter als ohne Clustering. Im Falle der Clustering wie analog zu Abbildung 5.38 mit den oberen bzw. unteren Ecken in jeweils einem Cluster, ist die Verteilung deutlich passender und sollte aufgrund der typischen HMI Inhalte so gewählt werden, wenn eine Clustering

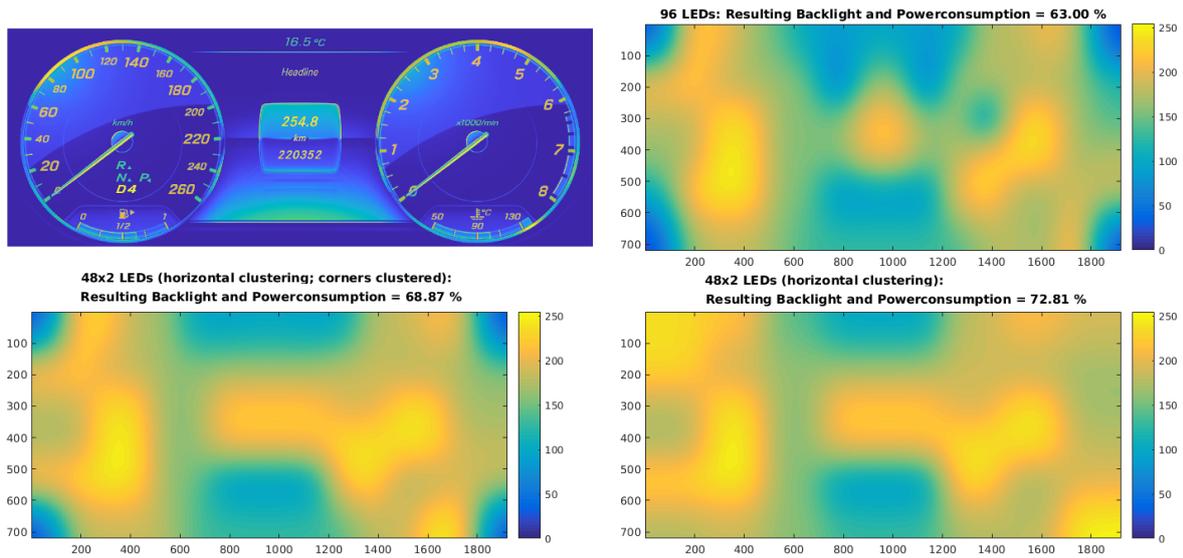


Abbildung 5.39.: Anwendung der Clusteringmethode auf einer 96-LED BLU. Oben links: Testbild in Falschfarben (aufgrund von Non-Disclosure Agreement (NDA, Vertraulichkeitsvereinbarung)). Oben rechts: Local-Dimming ohne Clustering. Unten links: Clustering nach Konzept aus Abbildung 5.38 mit *Corner-Clustering*. Unten rechts: Clustering direkter horizontaler Nachbarn.

erwünscht ist. Das Powersaving ist ebenfalls höher als im Falle der reinen horizontalen Clustering der direkten Nachbarn. Der Schwarzwert in den vier Ecken ist um Größenordnungen besser. Somit wird die Wertigkeit signifikant erhöht.

Diese Möglichkeit zur Nutzung des Clustering ist in der entwickelten Technologie berücksichtigt und kann somit flexible ohne Änderung des Hardwaredesigns durch Parameteranpassung realisiert werden.

Im nächsten Kapitel werden die wichtigsten in dieser Arbeit verwendeten Prototypen und FPGAs sowie der Hardware-Aufbau des Systems vorgestellt und die neue Systemarchitektur des entwickelten Local-Dimming Algorithmus eingeführt.

## 5.6. Neue Local-Dimming Systemarchitektur und Implementierung

In den vorherigen Abschnitten wurden Methoden vorgestellt, um eine Realisierung eines Local-Dimming Algorithmus in Hardware für die Anwendung im Automobil kostengünstig zu ermöglichen und eine gute visuelle Qualität zu erreichen. Insbesondere wurde die Methode der LSF Modelle eingeführt, um den benötigten Speicher drastisch zu reduzieren und die Pixelorderstruktur konzeptionell neu entworfen. Um die Funktion dieser neuartigen Konzepte zu überprüfen, wird in den kommenden Abschnitten die Realisierung in Hardware vorgestellt. Außerdem werden die verwendeten Prototypen und deren Aufbau präsentiert. Für den weiteren Ablauf wird in Abbildung 5.40 schematisch in einem Diagramm der Local-Dimming Algorithmus basierend auf den neu eingeführten Konzepten vorgestellt. Im Vergleich ist der SoA in Abbildung 3.24 auf Seite 62 abgebildet. Der Implementierung basiert dabei auf einer Architektur mit Pixelpipeline, sodass jeder Pixel mit einem (synchronen) Takt der *Pixelclock* durch die *Register* weiter geschoben wird. Die gesamte

Local-Dimming Implementierung wird auch als *Local-Dimming-Prozessor* bezeichnet und stellt auch die IP dar, welche zu potenziellen Anwendern, übertragen werden kann. In der unteren Abbildung ist eine Dreiteilung des Algorithmus zu erkennen. Wie in Abschnitt 3.5.1 eingeführt, teilt sich der Algorithmus in die Phasen *Preprocessing*, *Optimizer* sowie dem *Postprocessing* auf. Die in dieser Arbeit neu eingeführten Konzepte wurden größtenteils in den grün markierten Blöcken implementiert. Der Bilddatenfluss (in Hardware die Pixelpipeline) ist in rot dargestellt, während die LED-Werte mit grünen Pfeilen repräsentiert werden. Wie bereits erläutert, werden die neuen LSF-Modelle für die Berechnung der LED-Werte benötigt, um die zu jeder Condensierzelle gehörenden Einflüsse der  $5 \times 5$  Nachbar-LEDs zu bestimmen. Die darauf aufbauende Berechnung des virtuellen Backlights ist ebenfalls auf die Modelle angewiesen. Analog der in Abbildung 5.29

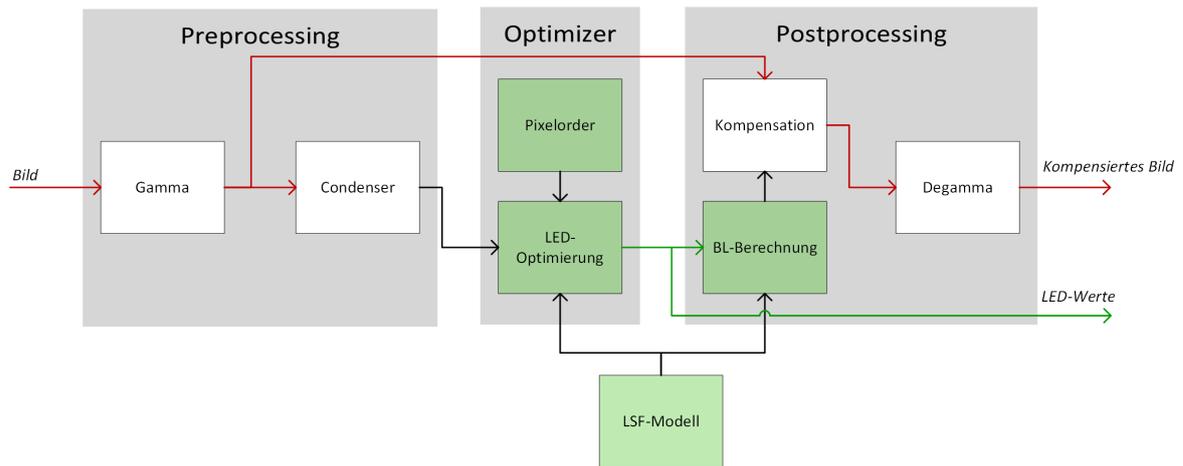


Abbildung 5.40.: Datenflussdarstellung des zu implementierenden Algorithmus. Grün hinterlegt sind die Blöcke in denen die neuen Konzepte integriert sind.

(Seite 157) dargestellten Stützstellen, wird für jeden Sample-Point das Backlight unter Berücksichtigung *aller* LEDs berechnet. Auf die einzelnen Abschnitte wird in Abschnitt 5.6.2 genauer eingegangen.

Die hier genutzten Direct-Lit BLUs wurden in Kapitel 4 vorgestellt. Für die automotive ICD Prototypen kamen die BLUs mit den LSFs aus Abbildungen 4.5, 4.6 und 4.18 zur Anwendung. Die Querschnitte der genutzten LSFs sind in Abbildung 5.41 abgebildet. Der violette Verlauf zeigt hier ein Referenzgerät, welches dem *Hisense TV* aus Abschnitt 4.1.2 entspricht. Anhand der Übersicht wird der Unterschied zwischen den einzelnen LSFs deutlich. Dies wurde bereits in Kapitel 4 hinreichend diskutiert.

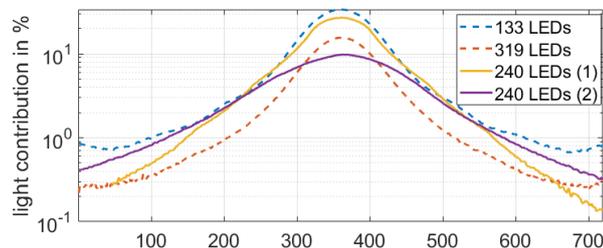


Abbildung 5.41.: Querschnitte der hier verwendeten BLUs. Entnommen aus [SAX20].

### 5.6.1. Systemkonzept und Aufbau

Die vorgestellten Konzepte wurden auf realer Hardware realisiert. Aufgrund starken Interesses verschiedener Automobilzulieferer und OEMs wurde auch Hardware von diesen Projektpartnern beigesteuert. Insbesondere konnten von *Continental* und *Nichia* LC-Panels und auch die BLUs zur Verfügung gestellt werden, um die hier entwickelte Technologie zu validieren und zu testen.

Zwei Prototypen sind in Abbildung 5.42 als Gesamtsystem abgebildet. In Abbildung 5.42a ist das FPGA-Board (rot) mit einer grünen „Adapter“ PCB gut zu erkennen. Rechts unten ist ein *Raspberry Pi Zero* als Bildquelle in Verwendung. Das angezeigte Bild auf dem Displays ist ein Teil eines internen Test- bzw. Demovideos von *Continental*, welches dem LME zur Verfügung gestellt wurde. Dahinter befindet sich eine BLU mit 240 LEDs, deren Kathoden und gemeinsame Anoden über das weiße Flachbandkabel mit dem LED-Treiber (Rückseite grüne PCB) verbunden sind. In Abbildung 5.42b ist ein ähnlicher Aufbau, jedoch basierend auf einem anderen FPGA-Board (Genesys2), jedoch mit demselben FPGA-IC dargestellt. Das schmale Flachbandkabel dient zur SPI-Kommunikation mit den LED-Treiber.

Die zum Einsatz genutzten LC-Panels (12,3 Zoll) haben eine Auflösung von  $720 \times 1920$  bei einer Farbtiefe von 8 Bit pro Farbkanal. Es ergibt sich eine Pixelclock der Frequenz von  $f_p = 98,42688$  MHz.

Der eigentliche Local-Dimming Algorithmus, welcher im FPGA realisiert ist, wird zwischen Bildquelle (*Source*) und Display (*Sink*) in den Bilddatenpfad integriert. In Abbildung 5.43 ist schematisch der Aufbau des Gesamtsystems dargestellt. Dabei sind die Bilddatenpfade (*Pixelpipeline*) in Rot, während die Übertragungspfade der LED-Werte in Blau dargestellt wird. Im Folgenden werden die dargestellten Komponenten zum besseren Verständnis des Aufbaus näher erläutert.

#### Bildquelle und Kommunikation

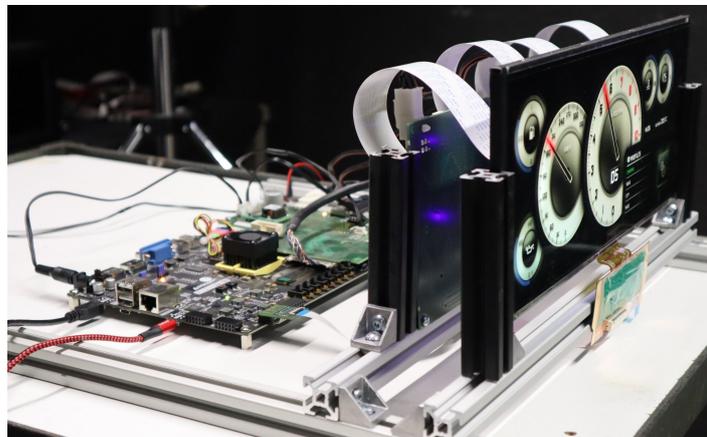
Dieser Block repräsentiert die Bildquelle sowie das für die Kommunikation und Parameteränderung notwendige UART Interface. Im Allgemeinen kann diese Aufgabe von einem einzigen Computer mit entsprechenden Interfaces übernommen werden. Aufgrund der Flexibilität wurde meistens jedoch die Bildquelle autark betrieben und kann beispielsweise aus einem *Raspberry Pi Zero* bestehen. Das Bildsignal wird über den HDMI, also über den TMDS Standard übertragen. Dazu werden 4 differenzielle Lanes (R,G,B sowie Pixelclock) benötigt. Die EDID Information wird über den I2C Bus ausgelesen, um das benötigte Timing zu verwenden. Die Kommunikation über UART wird von einer am LME entwickelten Java-GUI durchgeführt. Alternativ können die entsprechenden Klassen auch in Matlab instantiiert und genutzt werden.

#### Adapter PCB mit FMC

Diese PCB ist für die Übertragung der Bilddaten ins FPGA sowie aus dem FPGA zuständig. Zu diesem Zweck enthält sie eine HDMI-Buchse, welche die TMDS Signale an den IC *TFP141* [Tex16] durchleitet. Dieser IC wandelt das serialisierte HDMI-Signal in einen parallelen Bilddatenstrom (RGB parallel) um. Somit verlassen hier 28 Single-Ended Signale den IC. Es besteht aus Bilddaten ( $3 \cdot 8$ ) sowie den zugehörigen Sync-Signalen (*DE*, *vsync*, *hsync*) sowie der Pixelclock *p\_clk*. Somit entsteht hier eine zur Pixelclock synchrone Pixelpipeline. Auf dem PCB ist auch die EDID Information in einem EEPROM gespeichert und direkt mit dem I2C-Bus über die HDMI an die *Source* angeschlossen. Dieser EEPROM kann auch im FPGA realisiert werden, erleichtert jedoch so die Bedienung des Prototyps, da die Bildquelle den EEPROM auslesen kann, ohne dass das FPGA bereits konfiguriert ist. Dadurch wird das Vorhandensein der Pixelclock vor der FPGA-Konfiguration garantiert.



(a) Ein Prototyp basierend auf einem Inveium Board mit Xilinx Kintex 7 FPGA. Ein Raspberry Pi Zero dient hier als Bildquelle.



(b) Weiterer Prototyp basierend auf einem Genesys2 Board mit Xilinx Kintex 7 FPGA. Zu sehen ist das *blaue* Backlight, welches aus einigen Öffnungen aus der BLU rückseitig leuchtet.

---

Abbildung 5.42.: Prototypen mit FPGA-Board und LC-Panel auf 240er BLUs.

Weiterhin sind auf diesem PCB 2 DC/DC Wandler verbaut, welche 12V auf 3,3V konvertieren. Diese Spannungsversorgung dient den LEDs in der BLU als gemeinsame Anode.

### Xilinx Kintex 7 FPGA

Das FPGA beinhaltet die Implementierung/IP des Algorithmus (siehe Abbildung 5.40) und liefert die Bilddaten über FPD welches auf dem *physikalischen Layer* LVDS basiert. Dazu werden die parallelen RGB-Daten im FPGA im Verhältnis 1:7 serialisiert und auf 4 differentielle Lanes verteilt, sodass pro Pixelclock insgesamt 28 Bit übertragen werden. Die Clock selbst wird ebenfalls auf einer eigenen Lane übertragen. Diese Signale werden über den FMC zurück zum *Adapter PCB mit FMC* geroutet und von dort aus über Kabel an das *TCON PCB* geleitet. Zusätzlich zu den Bilddaten werden die berechneten LED-Werte per SPI Bis vom FPGA ausgegeben. Dazu werden die Signale MIS0, MOSI, SPI\_clk sowie LAT genutzt. Letzteres dient den LED-Treibern als Latch-Signal um die anliegenden Werte aus den Schiebe-Registern zu übernehmen.

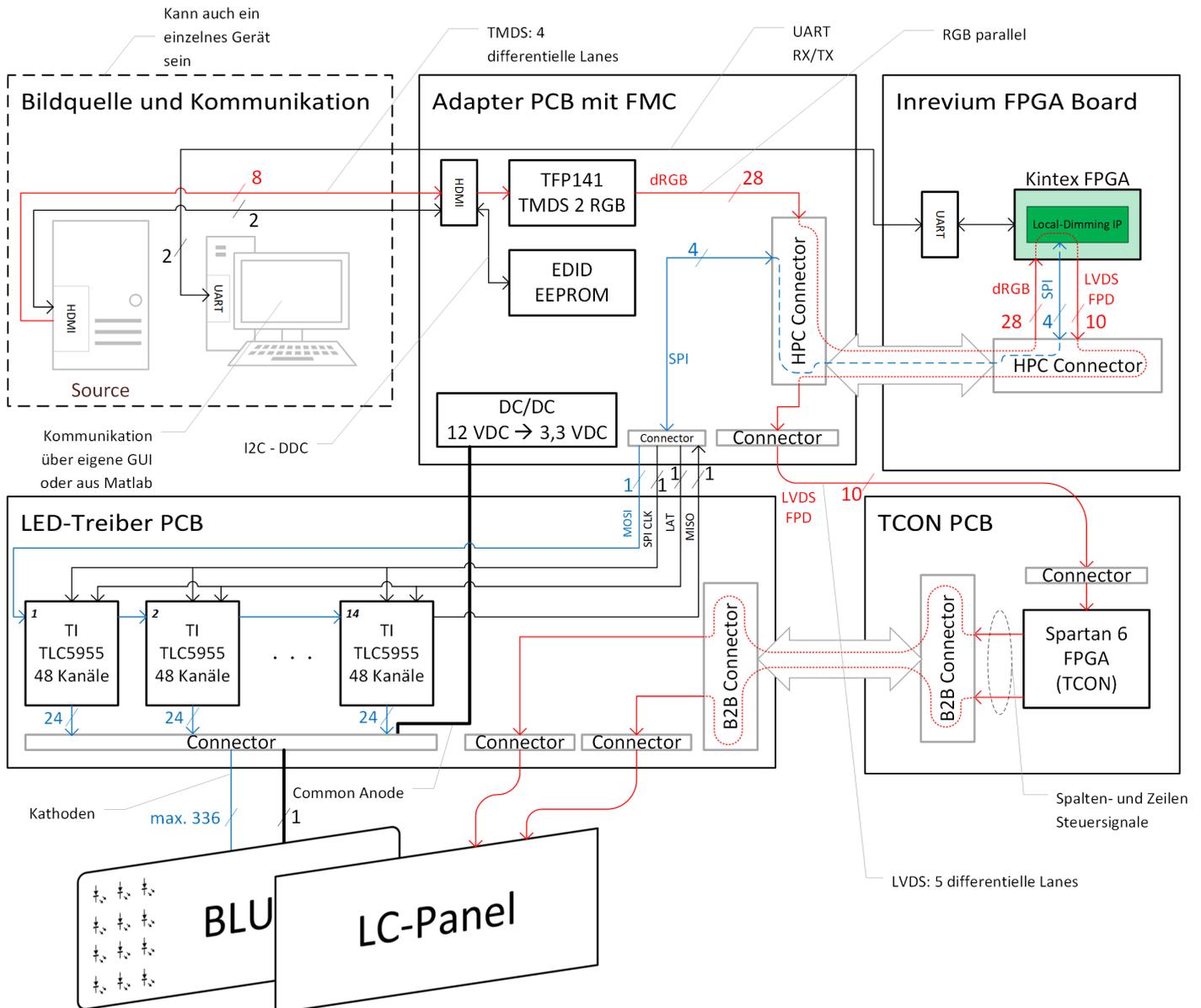


Abbildung 5.43.: Schematischer Aufbau und Konzept des Local-Dimming Prototyps. Der Bilddatenpfad ist in Rot, die Übertragung der LED-Werte ist in Blau dargestellt. Die eigentliche Local-Dimming IP ist grün hinterlegt.

### TCON PCB

Das TCON PCB beinhaltet einen Spartan 6 FPGA, welcher als TCON die Aufgabe hat, die Steuersignale für das LC-Panel zu generieren. Der TCON bekommt eingangsseitig die Bilddaten per FPD und steuert ausgangsseitig die TFTs der LC-Matrix an. Die Signale werden dazu auf das LED-Treiber PCB geroutet und von dort aus mit Flachbandkabeln an das Panel angeschlossen. Die Funktion des TCONs wird von dem Lieferanten des LC-Moduls realisiert.

## LED-Treiber PCB

Diese Platine ist mit 14 Treibern von *Texas Instruments* des Typs *TLC 5955* bestückt [Tex14]. Dieser Treiber-IC hat 48 einzelne Kanäle, die jeweils einen LED-Strang mit PWM ansteuern können. Die PWM-Auflösung beträgt dabei 16 Bit und ist ausreichend hoch. Zwei Kanäle werden paarweise kurzgeschlossen, sodass pro IC maximal 24 LEDs angesteuert werden können. Somit ergibt sich eine Obergrenze von 336 LEDs für diese PCB.

Für die Übertragung der berechneten LED-Werte sind die TLC5955-Chips so angeschlossen, dass eine *Daisy-Chain* entsteht. Das heißt, die Werte die über den SPI Bus übertragen werden, müssen durch die einzelnen ICs durch die Schieberegister der Treiber geschoben werden. Dazu wird der serielle Datenausgang jeweils mit dem Dateneingang des Folgechips verbunden. In Abbildung 5.44 ist diese Verschaltung im Detail dargestellt. Um alle Kanäle ansteuern zu können, müssen insgesamt  $14 \cdot (16 \text{ Bit} \cdot 48 + 1) = 10766$  Bit übertragen werden. Die SPI-Clock Frequenz wird zu 8 MHz festgelegt. Dadurch dauert eine Übertragung 1,35 ms.

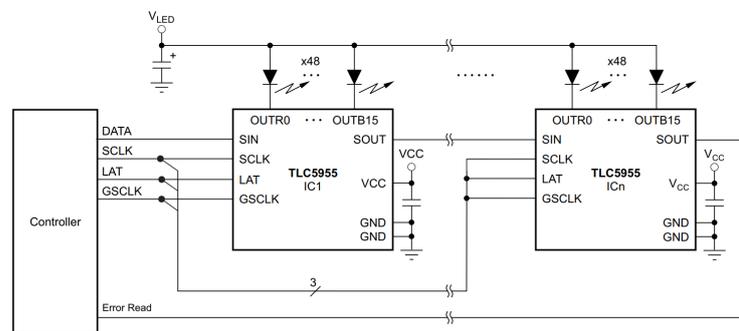


Abbildung 5.44.: Verschaltung der TLC-Treiber als *Daisy-Chain*. Der Controller ist in diesem Fall das Kintex 7 FPGA. Entnommen aus [Tex14].

## 5.6.2. Integration und Implementierung

In diesem Abschnitt wird näher auf die Hardwareimplementierung eingegangen. Hier ist anzumerken, dass der gesamte Algorithmus in Matlab entwickelt und für Validierungszwecke implementiert wurde, um den Videobetrieb testen zu können und die korrekte Umsetzung des Hardware-Designs zu garantieren. Zunächst wird auf die sogenannte *Offline-Prozessierung* eingegangen.

### 5.6.2.1. Implementierung einer *Offline-Prozessierung*

Im frühen Entwicklungsstadium des Algorithmus wurde für eine effiziente Entwicklung und Integration neuer Erkenntnisse und Methoden, das Verfahren in Matlab implementiert, sodass die Testbilder *offline* prozessiert werden konnten. Diese Ergebnisse können für Simulationen und später zur Validierung genutzt werden.

Um die Ergebnisse auf der echten Hardware anzeigen zu können, wird ein Hardwaredesign erzeugt, welches die LED-Werte zugeführt bekommt und diese an die LED-Treiber senden kann. Ein Vorteil eines solchen Systems ist, dass bereits am Anfang die Funktionalität des Hardwareaufbaus überprüft werden kann. Ein weiterer Vorteil ist, dass mithilfe der in diesem Abschnitt vorgestellten Übertragung der LED-Werte in den Bilddaten, beliebige LED-Daten übertragen und an den LED-Treiber gesendet werden können, ohne dass ein

zusätzliches Kommunikationsinterface zu diesem Zeitpunkt notwendig wäre. Somit können die LSFs zeitnah vermessen werden. Das entwickelte Local-Dimming System kann folglich *offline* in Matlab ausgeführt und die LED-Lösungen in das Bild gespeichert werden. Der Output ist folglich ein in einem Bildformat (e.g. png) gespeichertes Bild, welche auch die neben dem kompensierten Bilddaten auch die berechneten LED-Werte enthalten. Ein sogenanntes *Durchschleif-Design* im FPGA reicht das Bild über die Pixelpipeline an den TCON des LC-Panels weiter. Dabei werden die gespeicherten LED-Werte dekodiert und an den LED-Treiber gesendet. Damit kann die Offline-Prozessierung getestet werden.

Ist die Matlab Implementierung des Local-Dimming Algorithmus getestet und validiert, kann sie als Referenzimplementierung angesehen werden. Die Offline-Prozessierung erlaubt natürlich keinen Echtzeit-Betrieb. Um diesen zu ermöglichen wird im nächsten Schritt die Echtzeitprozessierung in Hardware implementiert, welche mit einer Hardware Description Language (HDL, Hardwarebeschreibungssprache) beschrieben wird und im FPGA auf der Pixelpipeline arbeitet.

### 5.6.2.2. Echtzeit/Online-Prozessierung

Nach erfolgreicher Inbetriebnahme der Hardware und Validierung der neu entwickelten Konzepte mithilfe der oben vorgestellten Methode, wurde der Algorithmus komplett in HDL implementiert. In Abbildung 5.40 auf Seite 170 ist das Blockdiagramm des Algorithmus abgebildet. Diese IP wird nun wie in Abbildung 5.43 angedeutet, in der Pixelpipeline im FPGA realisiert. Hier wird nun auf die Implementierung der neu entwickelten Konzepte eingegangen. Die Modulübersicht dazu ist in Abbildung 5.45 dargestellt. Für die Realisierung des RAMs zur Speicherung der berechneten Sample-Points sowie für den *Condenser*, wird jeweils pro Frame ein Speicher nur zum Beschreiben genutzt, während der andere parallel für die Berechnung ausgelesen werden kann. Dieses RAM-Swapping ist aufgrund des knappen Timings notwendig. Ein weiterer Vorteil ist, dass der RAM-Inhalt während der eines Bildes konstant bleibt. Zur Vorbereitung für höhere Pixelfrequenzen und einer ASIC Implementierung ist die Pixelpipeline mit der Frequenz  $f_p$  in zwei parallele Pipelines mit jeweils der Frequenz  $f_p/2$  geteilt. Diese Unterteilung in zwei *Pipes* wird im Modul *E/O-Splitter* durchgeführt („Even-Odd-Splitter“). Dazu wird aus der Pixelclock  $p\_clk$  eine Clock mit halber Frequenz in der bereits genutzten Phase-Locked Loop (PLL, Phasenregelschleife) erzeugt. Das Modul tastet dann mit der halben Frequenz jeweils zwei benachbarte Pixel (Even-Pixel, Odd-Pixel) im Pixelstream der vollen Pixelclock ab. Der inverse Weg wird durch das Modul *E/O-Combiner* durchgeführt. Bei einem Prototyp mit einer Auflösung von  $2160 \times 3840$  Pixel wurden sogar 4 parallele Pipes genutzt (*Consumer-TV*). In [Sch17] wurde dieses Prinzip bereits eingeführt, um die Pixelfrequenz reduzieren zu können. Dadurch kann der *kritische Pfad* [KC66], also der Signalpfad durch die Logik mit der maximalen Durchlaufzeit zwischen zwei Registern inklusive dem Leitungsdelay (*Net-Delay*), länger werden. Somit kann sich tiefere kombinatorische Logik zwischen zwei Registerstufen befinden, ohne dass die Verzögerungszeit zu hoch wird.

An dieser Stelle sei nochmal erwähnt, dass zwar die Funktion von ROMs (im Sinne von einem Speicher mit ausschließlich Lesefunktionalität) mit der HDL beschrieben werden, diese im FPGA und auch im ASIC mit dem vorhandenen SRAM realisiert sind. Die *Schreib*-Signale werden dauerhaft auf 0 gesetzt und die Inhalte bei der Konfiguration des FPGAs in die SRAM-Blöcke geschrieben.

Im Folgenden werden die Funktionen der hinzugefügten Module kurz erläutert. In Rot (Pfade) ist der Pixelstream mit der vollen Frequenz  $f_p$  dargestellt, in Braun (Pfade) die zwei Pipes mit  $f_p/2$ .



- *clk\_mux*: Nutzt physikalisch vorhanden Clock-MUXer auf dem FPGA um zwischen der Pixel- und der Systemclock umzuschalten. Aufgrund der zeitlichen Constraints (siehe auch Abschnitt 5.1.2) läuft die eigentliche LED-Berechnung als auch die Sample-Point-Berechnung mit der Systemclock *sys\_clk* mit einer Frequenz von  $f_s = 200$  MHz. Das Umschalten ist notwendig, da jeweils einer der RAM Blöcke mit der Pixelclock synchron getaktet werden muss, um Condenserwerte zum laufenden Bild zu berechnen und zu speichern als auch zum Kompensieren der Pixel. Die anderen beiden RAMs takten in dieser Zeit synchron zur LED-Berechnung (*Optimizer*) und wechseln beim bei einem neuen Frame.
- *Optimizer Top*: Eigentliche LED-Berechnungsmethode und Modelle aus dieser Arbeit. Mit der Systemclock getaktet.
  - *calc\_delta*: Berechnet für jede Condensierzelle den fehlenden Lichtbeitrag nach Formel (5.15) auf Seite 161 mithilfe der *Pixelorder* und der LSF-Modelle.
  - *increase-LEDs*: Berechnet mit Information aus *calc\_delta* nach Formel (3.5) den neuen LED-Wert der aktuell zu erhöhenden LED.
  - *calc\_bl\_SPs*: Berechnet mithilfe der LSF-Modelle das resultierende Backlight nach der Berechnung der LED-Werte.
  - *lsf\_decode*: Das Modul zum Bestimmen des modellbasierten Einflusses nach neuer Methode. Enthält auch die Modelle im ROM mit Modellspeicher nach Formel (5.11). Aufgrund der Wichtigkeit für den Algorithmus wird auf dieses Modul weiter unten detaillierter eingegangen.

### 5.6.2.3. LSF-Decoder für *Optimizer Core* und Backlight Berechnung

#### Das *LSF-Decoder* Modul

Eines der wichtigsten Module, welche für die Implementierung der neuartigen LSF-Modelle ist das *LSF-Decoder* Modul. In Abbildung 5.45 wurde bereits angedeutet, dass dieses Modul sowohl für die Berechnung der LED-Werte als auch für die Berechnung des finalen, resultierenden Backlights im Modul *calc\_bl\_SPs* verwendet wird. Die Aufgabe dieses Moduls ist die Bestimmung des Einflusses, welche in den 9 Modellen gespeichert wird und ermöglicht die Abdeckung der LSFs aller LEDs durch diese Modelle. Es soll also nach einer bestimmten Durchlaufzeit, taktweise liefern:

$$\text{LSF\_Decoder}(\text{LED}_{\text{num}}, \text{SP}_x, \text{SP}_y) = \check{L}_{(\text{SP}_y, \text{SP}_x), \text{LED}_{\text{num}}} \quad (5.17)$$

Dazu wird auch eine Adresslogik benötigt, die im Anschluss vorgestellt wird. Das Modul soll also zu einem Pixel oder Sample-Point  $p$  den Einfluss einer LED  $l$  in Abhängigkeit von den SP-Koordinaten und den Modellen liefern (vgl. beispielsweise Formel (5.6)) auf Seite 146. Zudem werden in diesem Modul die Mapping-Funktionen, wie in Abbildung 5.20 visualisiert, in einem ROM (*LUT\_LED\_num\_to\_model*) gespeichert. Dieses ROM liefert also zu einer LED-Nummer  $l$  (entspricht der Adresse) folgendes Ergebnis, welches eine Breite von 8 Bit aufweist.

$$\text{LUT\_LED\_num\_to\_model}(l) = \left( \underbrace{m_{nr}}_{4 \text{ Bit}}, \underbrace{m_y}_{1 \text{ Bit}}, \underbrace{m_x}_{1 \text{ Bit}}, \underbrace{t_b}_{1 \text{ Bit}}, \underbrace{l_b}_{1 \text{ Bit}} \right) \quad (5.18)$$

Die Daten von links nach rechts, sind wie folgt zu interpretieren:

$l \mapsto (\text{Modell-Nummer, gespiegelt an } y\text{-Achse, gespiegelt an } x\text{-Achse, obere Kante, linke Kante})$

Das zweite ROM, welches jeder LED  $l$  ihre  $x$ - und  $y$ -Position in SP Auflösung liefert, lässt sich folgendermaßen beschreiben:

$$\text{LUT\_LED\_num\_coordinates}(l) = (\underbrace{l_{\text{pos}_y}}_{6 \text{ Bit}}, \underbrace{l_{\text{pos}_x}}_{8 \text{ Bit}}) \quad (5.19)$$

Die Gesamtbreite von 14 Bit ergibt sich aus der maximalen SP-Auflösung in horizontaler und vertikaler Richtung. Nach Formel (5.9) können die Positionen bestimmt werden.

Das Blockdiagramm des *LSF-Decoders* ist in Abbildung 5.46 auf Register Transfer Level (RTL, Registertransferebene) Ebene dargestellt. Da dieses Modul zeitkritisch ist, muss die Einflussbestimmung in einer Pipeline bestimmt werden, sodass nach einem Offset, jeden Takt der nächste Einfluss ausgegeben wird. Die Pipelinestufen sind in der Abbildung ebenfalls visualisiert. Nach Anlegen der Eingangsdaten zum Zeitpunkt  $t_0$  liefern die oben genannten ROMs im Folgetakt die entsprechenden Inhalte ( $t_1$ ). Diese durchlaufen dann die Adressierlogik *calc\_LSF\_address* in  $x$  Takten, sodass die korrekte Adresse zum Adressieren der Modellspeichers ( $t_{1+x}$ ) anliegt und schließlich nach insgesamt  $t_{1+x+1}$  Takten zur Verfügung steht. Zu Beachten ist, dass der *LSF-Decoder* aufgrund des Designs mit Pipelining, *jeden* Takt neue Werte einlesen kann. Die Register sind dazu da, die Daten in der Pipeline synchron zu halten und dienen als *Delay* Elemente. Auf der rechten Seite sind die ROMs abgebildet, welche die 9 Modelle ( $M_i^{\text{SP}'}$ ) speichern. Dabei wird, wie Formel (5.11) auf Seite 158 berechnet, zwischen der vollen räumlichen SP und der reduzierten SP-Auflösung unterschieden. Die zweidimensionalen Modelldaten werden im ROM so linearisiert gespeichert, dass sie die einzelnen Modelle zusammenhängend sind, sodass nur die Offsets zwischen den Modellen bei der Adressierung beachtet werden müssen. Die Modell-SPs der vollen SP-Auflösung werden somit getrennt von den reduzierten Modell-SPs in einem weiteren ROM gespeichert. Die Bitbreiten der Adressleitungen bestimmen sich durch die Anzahl der Einträge für die volle SP-Auflösung zu  $\lceil \log_2(49^2 \cdot 9) \rceil = 15 \text{ Bit}$  und für die reduzierte Auflösung zu  $\lceil \log_2(40 \cdot 25 \cdot 9) \rceil = 14 \text{ Bit}$ .

### Adressierung des Modellspeichers im *LSF-Decoder*

Für die Adressierung der beiden ROMs sei zur Veranschaulichung auf die Lage der Modelle in Abbildung 5.17 auf Seite 145 verwiesen. In Codeauszug C.1 auf Seite 249 ist ein Teil der Adressberechnung des *LSF-Decoders* in Matlab dargestellt. Es wird sofort ersichtlich, dass diese Berechnungsmethode so wie sie angegeben ist, in Hardware mehrere Takte benötigen würde. Die Logik wäre durch die verschachtelte `if-elseif-...` Struktur sehr tief und die Durchlaufzeit entsprechend hoch, sodass ein Takt der Systemclock nicht mehr ausreicht. Aus diesem Grund muss diese Adressberechnung (für *calc\_LSF\_address*) in eine Pipeline taugliche Form für die Verilog-Implementierung überführt und optimiert werden.

Zur Vorbereitung auf die Einführung des Pipelining für die Modelladressierung, werden die verschiedenen mögliche Fälle untersucht. Als Beispiel sind die Kombinationen der eingeführten *Flags* für das Modell 9 zur Berechnung der  $x$ -Komponente ( $x$ -Koordinate des Modells) in Tabelle 5.6 dargestellt. Die dritte Spalte gibt an, ob der aktuell betrachtete Sample-Point rechts oder links von dem LED-Zentrum der betrachteten LED liegt (es wird auch die vertikale Spiegelung des Modells berücksichtigt). Zusammen mit Codeauszug C.1

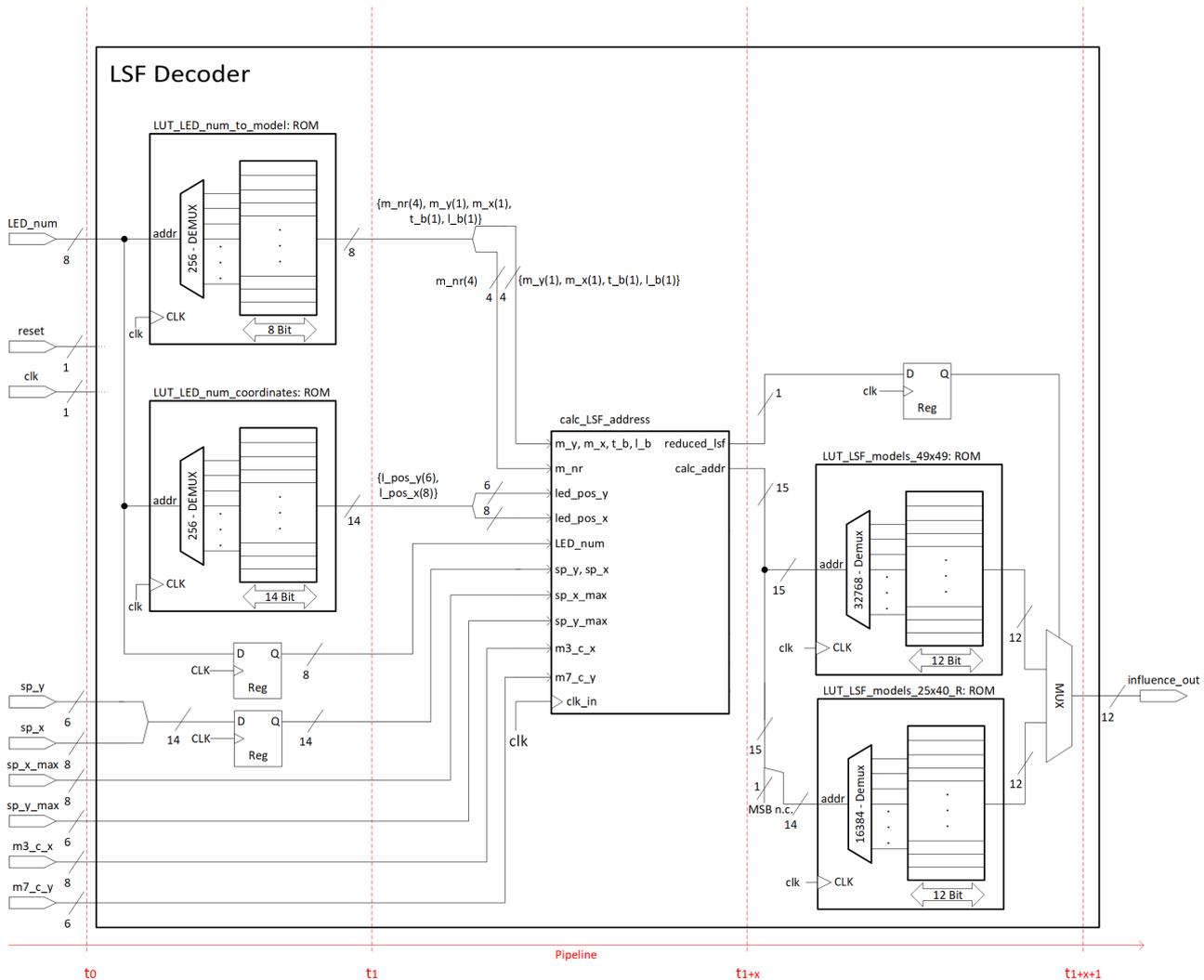


Abbildung 5.46.: LSF-Decoder Modul mit Pipelining in RTL Darstellung.

(Seite 249) ergeben sich 4 verschiedene Berechnungsfälle für die Bestimmung der x-Komponente. Diese ergeben sich damit zu den Vorschriften wie in Codeauszug 5.1 dargestellt.

Da die Modelle  $M_i^{SP'}$  mit  $i \in \{1, 2, 4, 5\}$  (vgl. Abbildung 5.20, Seite 147) nur gespiegelt, aber nicht verschoben werden, gibt es für diese auch weniger Fallunterscheidungen zu beachten. Die Berechnung der y-Komponente sowie die möglichen Adressen für die übrigen Modelle ergeben sich analog hierzu. In der Tabelle zeigen die grün hinterlegten Zeilen an, dass für manche Fälle die Berechnungsvorschrift gleich bleibt. So spielt es z.B. keine Rolle, ob der Sample-Point links oder rechts des LED-Zentrums liegt, wenn die aktuell betrachteten LED welches durch ein Modell repräsentiert wird, sich am Rand befinden. In diesem Fall wird der Wert wie in Formel (5.6) auf Seite 146 verwendet.

Modell 9			
mirror_y	left_border	SP_x > LED_SP_x	lsf_addr_x=
0	0	0	lsf_addr_x_4
0	0	1	lsf_addr_x_3
0	1	0	lsf_addr_x_2
0	1	1	lsf_addr_x_2
1	0	0	lsf_addr_x_3
1	0	1	lsf_addr_x_4
1	1	0	lsf_addr_x_1
1	1	1	lsf_addr_x_1

Tabelle 5.6.: Zuordnung verschiedener Modell-Adressen in Abhängigkeit der LED und SP-Lage für das Modell 9 (x-Richtung).

```

1  lsf_addr_x_1 = sp_resolution_x - sp_position_x;
2  lsf_addr_x_2 = sp_position_x;
3  lsf_addr_x_3 = lsf_model13_led_center.sp_x + sp_position_x - current_led_params.sp_x;
4  lsf_addr_x_4 = lsf_model13_led_center.sp_x + current_led_params.sp_x - sp_position_x;

```

Codeauszug 5.1: Verschiedene Vorschriften zur Adressberechnung für die Modelladressierung.

Das Blockdiagramm der vorgestellten Adressierlogik (*calc\_LSF\_address*) innerhalb des *LSF-Decoders* ist in Abbildungen 5.47 und 5.48 auf RTL Ebene abgebildet. Aufgrund der Größe des Diagramms wurde es in mehrere Teile zerlegt. Die gestrichelt dargestellten Ausgänge deuten auf die Fortsetzung im nächsten Bildausschnitt an. Die einzelnen Pipeline-Stufen sind wie oben durch rote Linien mit entsprechenden Zeitdarstellungen  $t_x$  dargestellt. Es wurde bereits gezeigt, dass je nach Konstellation der Mapping-Flags, verschiedene Berechnungsvorschriften zu tragen kommen. Zusätzlich werden noch die LED-Koordinaten in gespiegelter und ungespiegelter Form benötigt. Mit diesen Vorschriften kann die erste Pipeline-Stufe innerhalb der *calc\_LSF\_address* Moduls implementiert werden, welche für x- bzw. y-Koordinaten parallel arbeiten. Diese erste Stufe  $t_0$  bis  $t_1$  liefert die benötigten Adressen in einer Vorberechnung, welche in der darauf aufbauenden Stufe zur weiteren Entscheidungsfindung genutzt werden. Diese Berechnungen finden in den Untermodulen *precalc\_lsf\_addr\_1D* statt, welche für x- und für y-Koordinaten jeweils instantiiert sind, da die Berechnungen der Adressen analog durchgeführt werden können. Dieses Untermodul ist als Verilog-Code in Codeauszug C.3 (Seite 251) dargestellt.

Nach der weiteren Stufe, durchlaufen diese vorberechneten Adressen die beiden Module *conditional\_case*, welches ebenfalls doppelt instantiiert werden (mit  $\text{BIT\_DEPTH} = 8$  bzw.  $\text{BIT\_DEPTH} = 6$ ). Deren Beschreibung ist in einer HDL in Codeauszug C.4 (Seite 251) gezeigt. Die beiden Instanzen liefern also jeweils die x- und y-Koordinate innerhalb des Modells. Diese werden im weiteren Verlauf linearisiert (mithilfe eines *Multiplizierers*, analog zu Formel (4.1) auf Seite 77), sodass der Modellspeicher im übergeordnetem Modul mithilfe von Offsets ausgelesen werden kann. Beispielsweise entspricht der Offset für das Modell 9 dem Wert 19207. In Codeauszug C.2 auf Seite 250 sind die gespeicherten Offsets dargestellt. Diese werden ebenfalls in einem ROM gespeichert und beinhaltet  $2 \cdot 9 = 18$  Einträge. Außerdem berechnet dieses Modul das Steuersignal, um zwischen dem reduzierten und dem voll aufgelösten Modellspeicher zu wählen. Berechnet werden immer beide Adressen, sodass die Anzahl der Registerstufen nicht zusätzlich erhöht wird.

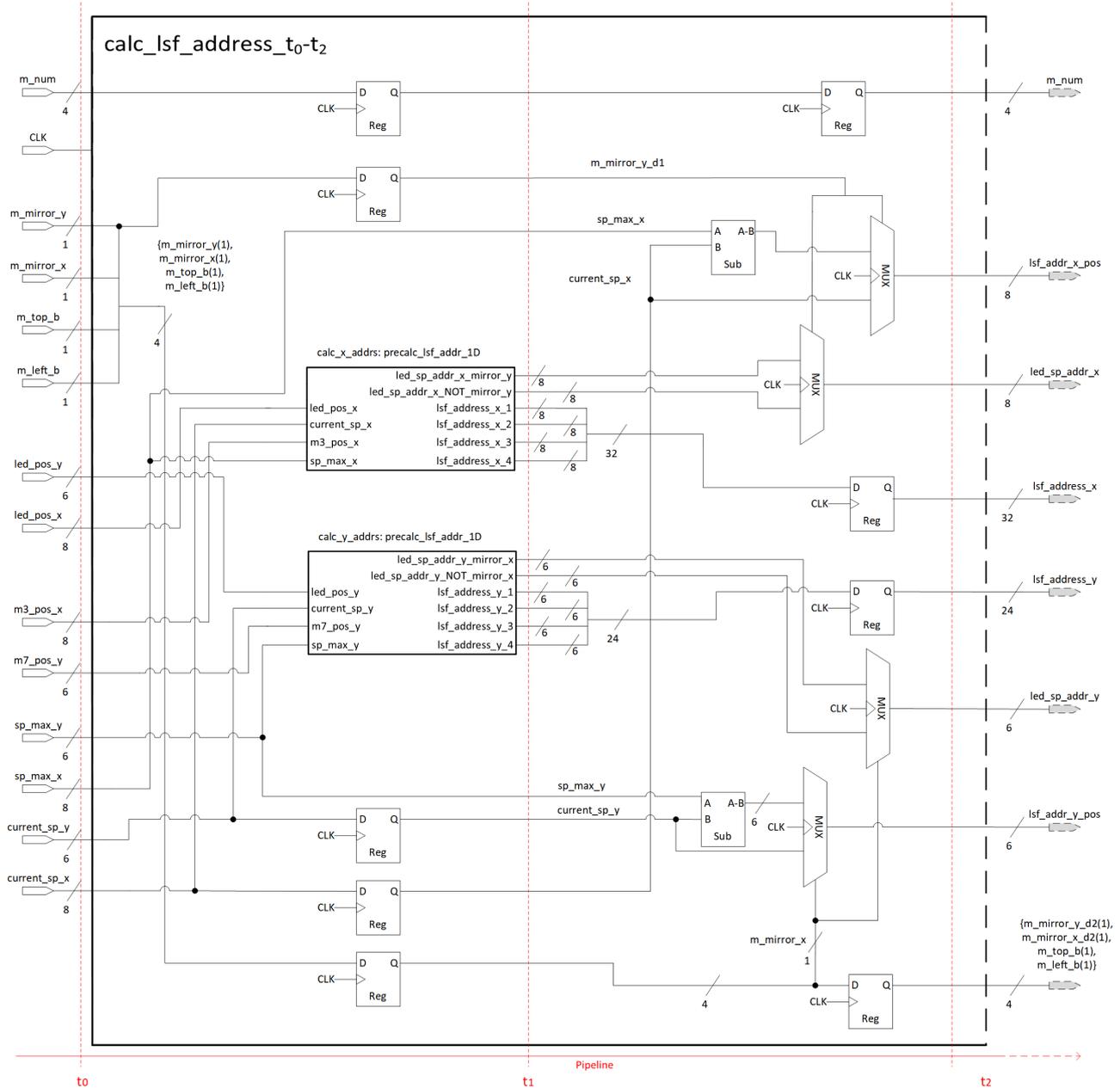


Abbildung 5.47.: Linke Hälfte des *calc\_LSF\_address* Moduls. Dargestellt sind die ersten beiden Registerstufen.

Insgesamt benötigt die Adressberechnung 15 Takte zur Berechnung der finalen Adresse des Modellspeichers. Die ganze Pipeline von dem Eingang bis zum Ausgang des *LSF-Decoders* besteht zusammen aus 17 Registerstufen. Diese sind in Abbildung 5.49 verdeutlicht. Die Signale *clk\_i* (Eingangsclock, hier 200 MHz), *SP\_y\_c\_i*, *SP\_x\_c\_i* (aktuelle Sample-Point x- bzw. y-Koordinaten) sowie *LED\_num\_c\_i* (aktuelle LED-Nummer) stellen die synchronen Eingänge des *LSF-Decoders* dar. Das unterste 12 Bit Signal *rel\_influence* stellt den Ausgang, also den Einfluss aus dem Modell dar. Die gelben Signale sind die Ausgänge der beiden ROMs aus der ersten Stufe des *LSF-Decoders*, welche die LED-Koordinaten und die Modell-Mappings liefern.

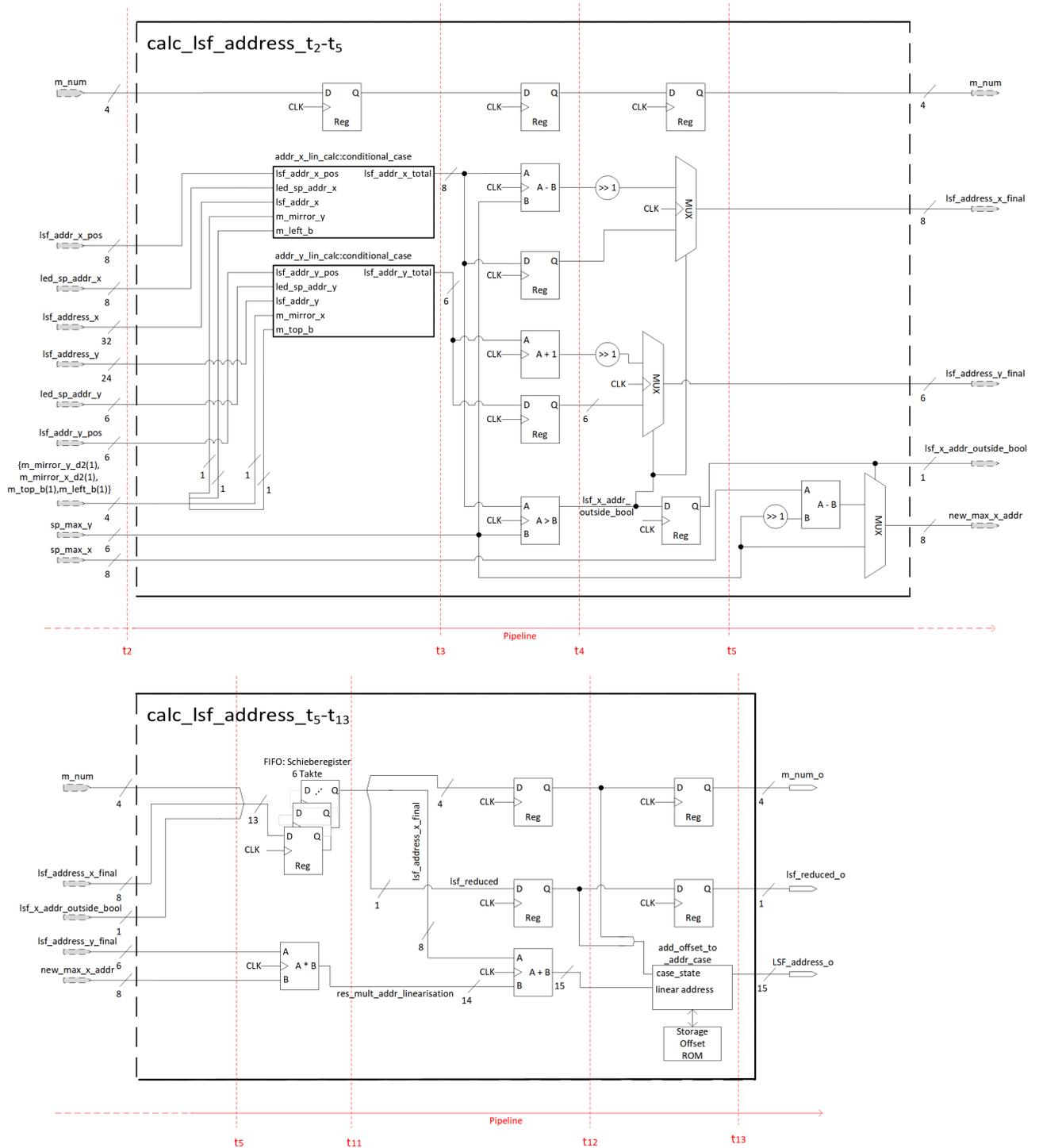


Abbildung 5.48.: Fortsetzung des Moduls mit Mittelteil und Ende. Das genutzte Schieberegister (als FIFO) wird genutzt, um die Daten mit dem Ausgang des Multiplizierers synchron zu halten ( $t_5 \mapsto t_{11}$ ).

Die rote Markierung zeigt dabei die zum markierten (gelbe Linie, gestrichelt) Start, die zugehörigen Werte innerhalb der Pipeline. Die grüne Markierung symbolisiert die zugehörigen Werte zu den Eingangssignalen

im Folgetakt. Es ist zu Erkennen, dass nach einem Delay von 17 Takten, die Einflüsse jeden einzelnen Takt am Ausgang anliegen. In diesem Beispiel bestimmt der Decoder den Einfluss (siehe Formel (5.17)) mit  $SP_y = 1$ ,  $SP_x = 2$ ,  $LED_{num} = 0$  zu dem 12 Bit Festkomma-Wert 1057 und entspricht 25,8%. Für die nächste LED ( $LED_{num} = 1$ ) im selben Sample-Point liefert das Modul den Wert  $636 \hat{=} 15,5\%$ . Bei den weiteren gezeigten Signalen, die hier nicht detailliert beschrieben wurden, handelt es sich um interne Signale innerhalb des *LSF-Decoder-Moduls*. Gut zu erkennen sind beispielsweise die parallel berechneten *lsf\_addr\_x/y* Adressen während der Adressberechnung.

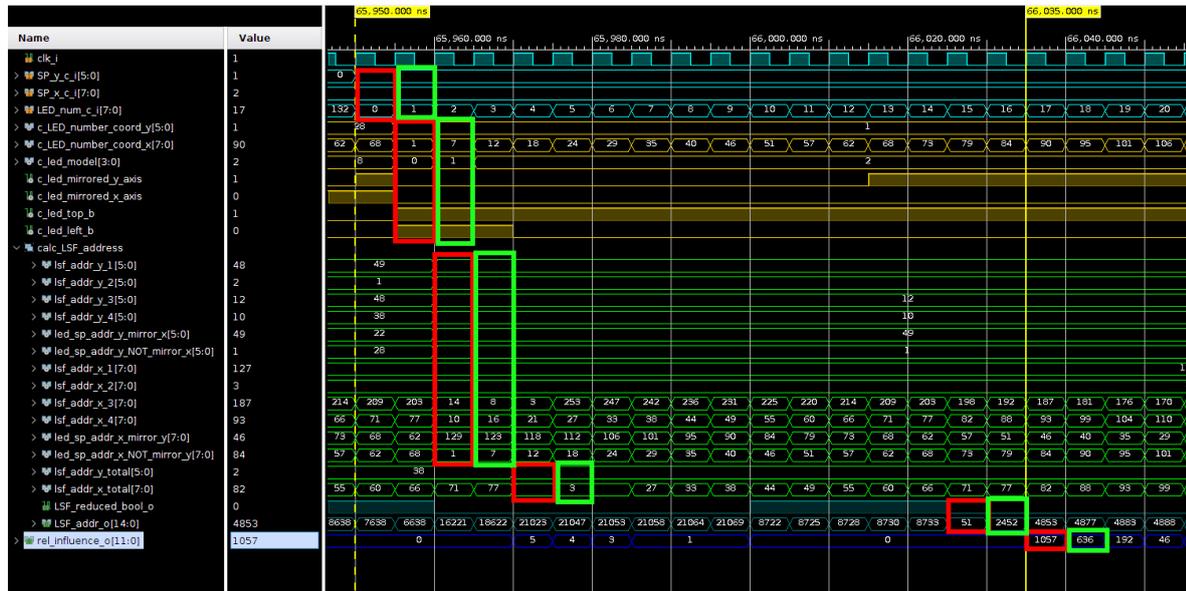


Abbildung 5.49.: Visualisierung der Pipeline des *LSF-Decoders* mit 17 Takten.

### Validierung des *LSF-Decoders*

Das vorgestellte Modul beinhaltet die neu eingeführten Modelle sowie die benötigte Adressierlogik um auf diese Modelle korrekt zugreifen zu können. In Abbildung 5.45 wurde ersichtlich, dass dieses Modul in mehreren Prozessierungsabschnitten genutzt werden muss.

Um die Funktion des entwickelten Hardwaremoduls zu garantieren, wird es gegen die Matlab Implementierung validiert. Die Resultate des Moduls müssen mit den Resultaten aus der Referenzimplementierung in Matlab übereinstimmen. Zudem muss die Berechnungs- bzw. Adressierungspipeline auf Funktion überprüft werden. Da ein manueller Vergleich aufgrund der großen Anzahl von Eingangskombinationen

$$(LED_{num}, SP_x, SP_y) \in \{0, \dots, N_L - 1\} \times \{0, \dots, N_{SP}^c - 1\} \times \{0, \dots, N_{SP}^r - 1\}$$

nur stichprobenartig geschehen kann, reicht es für eine Validierung in keinsten Weise aus. Folglich muss die Validierung automatisiert erfolgen. Dies geschieht unter Nutzung von sogenannten *Assertions*. Diese sollen beim Testen des Designs genutzt werden, um die Ausgänge des Verilog-Moduls mit den berechneten Ergebnissen aus der Matlab Implementierung auf Gleichheit zu vergleichen. Eine *Assertion* wird von Foster u. a. in [FKL06, S. 3 f.] sehr verständlich wie folgt definiert:

In general, an *assertion* is a statement about a design's *intended behavior* (that is, a property), which must be verified. Unlike design code, an assertion statement does not contribute in any form to the element being designed. Its sole purpose is to ensure consistency between the designer's *intention*, and what is *created*.

Da es in der Verilog HDL keine fertige *assertion* Funktion definiert ist, wird diese Funktionalität für die Hardwaresimulation durch ein eigenes Modul wie in Codeauszug 5.2, implementiert. Dieses Modul kann dann unabhängig von dem Unit Under Test (UUT, Prüfling) genutzt werden, um taktweise Signale miteinander zu vergleichen. Dieses *assert* beendet sofort die Simulation, wenn die zwei verglichenen Signale im selben Takt unterschiedlich sind und gibt in der Konsole die beiden ungleichen Signale, sowie den zugehörigen hierarchischen Namen des instantiierten Assert-Moduls aus. Somit können mehrere Instanzen für verschiedene Signale unterscheidbar genutzt werden. Ausgangssignale hat dieses Modul entsprechend seiner Funktion nicht.

```
1 module assert #(parameter DATA_WIDTH = 8)(
2     input clk,
3     input [DATA_WIDTH-1 : 0] test_v1,
4     input [DATA_WIDTH-1 : 0] test_v2);
5 always @(posedge clk)
6     begin
7         if (test_v1 != test_v2) // ignoriert X und Z Zustände
8             begin
9                 $display("v1: %d", test_v1) ;
10                $display("v2: %d", test_v2) ;
11                $display("ASSERTION FAILED in %m");
12                $finish; // Beendet die Simulation instantan
13            end
14        end
15    endmodule
```

Codeauszug 5.2: Ein *assert* Modul zur Verwendung in einer Verilog-Testbench.

Wie beschrieben müssen die Ergebnisse als auch die Berechnungspipeline des Moduls validiert und auf Funktion getestet werden. Dazu werden zwei verschiedene Methoden angewandt. Für die Überprüfung der Pipeliningfunktionalität wird das *LSF\_Decoder* Module als Unit Under Test (UUT, Prüfling) in einer Testbench instantiiert und in einer Simulation getestet. Das Konzept ist in Abbildung 5.50a dargestellt. Dazu wird ein Datensatz (gespeichert als Binärdaten) mit den möglichen Eingangskombinationen des *LSF\_Decoders* generiert und in Matlab mithilfe der Referenzimplementierung die entsprechenden Ausgaben (Einflüsse) berechnet, welche ebenfalls in einem zweiten Datensatz gespeichert sind. Dieser enthält also die Referenzwerte zu den Eingangsstimuli. Die Testbench nutzt nun den ersten Datensatz als Stimuli für die UUT (in HDL als ROM instantiiert). Dabei werden jeden Takt die nächsten Eingaben aus diesem ROM gelesen und an den Eingang des Modul (UUT) gelegt. Nach der Durchlaufzeit von 17 Takten wird der erste valide Einflusswert aus der UUT ausgegeben. Diese werden nun taktweise mit den Werten aus dem Referenz-ROM unter Beachtung des zeitlichen Unterschiedes von 17 Takten verglichen, in dem diese Signale dem *assert* Modul zugeführt werden. Somit wird mit dieser Methode nicht nur die Funktion der Pipeline getestet, sondern auch die erwarteten Ergebnisse validiert. Bei der ersten Unstimmigkeit wird die Simulation abgebrochen. Läuft diese durch, gilt die Funktion gegen Matlab validiert.

Bis hier wurde der *LSF\_Decoder* nicht im realen FPGA getestet. Jedoch muss auch die Funktion auf realer Hardware validiert werden. Die Überprüfung der Berechnungsergebnisse direkt im FPGA erfolgt wie in Abbildung 5.50b dargestellt. Dazu wird das Konzept *Hardware-in-the-Loop* genutzt. Eine gute Übersicht dazu bieten die Autoren Brayonov und Stoyanova in [BS19]. Die *Hardware* bzw. das UUT wird hier in den

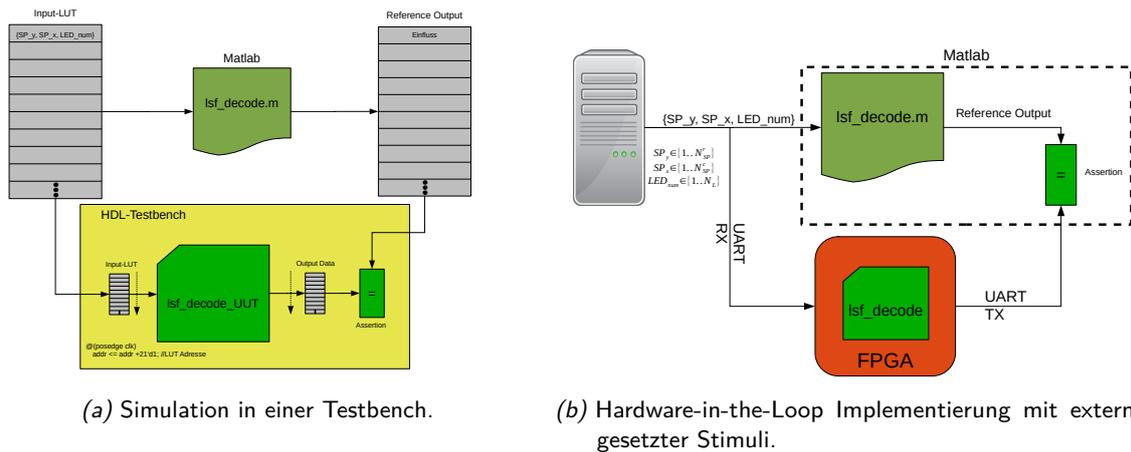


Abbildung 5.50.: Konzepte der Modulvalidierung des *LSF-Decoders*.

Datenpfad eingebunden. Dazu wird der *LSF\_Decoder* und das *UART* Modul im FPGA implementiert mit dem Ziel mit einem Computersystem zu kommunizieren. Die Kommunikation mit dem Design wird mithilfe eines am LME entwickelten Protokolls über UART realisiert. Es können immer nur 8 Bit Werte übertragen werden. Das Validierungsscript auf dem Computer generiert dabei eine Liste mit allen möglichen Stimuli für den Eingang des *LSF\_Decoder* ähnlich der Simulation aus Abbildung 5.50a. Dann wird daraus zufällig ein Wertetripel ausgewählt und aus der Liste entfernt. In Matlab wird basierend auf diesem Stimulus der entsprechende Einfluss berechnet. Außerdem werden diese Werte nacheinander zum FPGA übertragen, wo sie dann schließlich an die Eingänge des *LSF\_Decoder* gelegt werden. Nach 17 Takten mit der Systemclock mit  $f_s = 200 \text{ MHz}$  liegt das Ergebnis in einem 12 Bit Register. Dieses wird dann in ihre Most Significant Bits (MSBs) und Least Significant Bits (LSBs) unterteilt und entsprechend an den Computer gesendet (durch ein UART-Kommando getriggert) und muss auf diesem noch zu einer vorzeichenlosen, natürlichen Zahl zusammengesetzt werden. Sie wird schließlich mit dem Wert aus der Referenzimplementierung verglichen und muss ein identisches Resultat liefern. Der entsprechende Vergleich erfolgt mit einer Assertion in Matlab, welche im Fehlerfall die Validierung abbricht. Wird die komplette Stimuli-Liste ohne Abbruch durchgearbeitet, ist die Validierung abgeschlossen. Aufgrund des Protokolls und der genutzten Baud bzw. hier der Bitrate von  $115200 \text{ Bit s}^{-1}$  dauert das Übertragen eines Wertes zum FPGA inklusive des Overheads  $208,3 \mu\text{s}$  und liegt damit um Größenordnungen höher als die pipelinebedingte Durchlaufzeit des *LSF\_Decoder*. Somit kann diese hier vernachlässigt werden. Bei jedem Abfragen von Registern ist das korrekte Ergebnis am Ausgang des Moduls zu erwarten. Mit dieser Methode wurden also direkt in Hardware die Einflüsse aus dem Modellspeicher adressiert und ausgelesen sowie mit den Ergebnissen der Referenzimplementierung erfolgreich verglichen. Somit ist die gewünschte Funktionalität sichergestellt und das *LSF\_Decoder* Modul kann im Hardwaredesign des Local-Dimming Algorithmus eingesetzt werden.

#### 5.6.2.4. Funktion weiterer Module im *Optimizer* und Integration des *LSF-Decoders*

Oben wurde auf die Hardwarerealisierung des *LSF-Decoders* eingegangen. Hier wird nun das Design des hierarchisch übergeordneten Moduls, des *Optimizer\_Top* kurz erläutert. In Abbildung 5.45 (Seite 176) ist der grobe Aufbau in der Blockstruktur dargestellt, deren Module auf Seite 177 bzgl. ihrer Funktion erläutert wurden. In Abbildung 5.51 wird das resultierende Design des genannten Moduls als Übersicht in

einer abstrahierten Form in RTL Darstellung visualisiert. Zudem ist die Anbindung an den *außerhalb* des Moduls angesiedelten SRAM zur Speicherung der Condenserwerte und Backlight Sample-Points angedeutet. In der oberen linken Ecke ist eine Blackbox dargestellt, welche Steuerlogik enthält. Unter anderem befinden sich dort *Clock-Muxer*, welche zwischen der Systemclock (200 MHz) und vom Display (genauer: Display-Timing) abhängigen Pixelclock (hier 98 MHz) umschaltet. Zudem steuert die Logik die Adressierung und das Muxen zwischen den SRAMs bei Bildwechsel. Bei den Clock-Muxern handelt es sich um auf dem

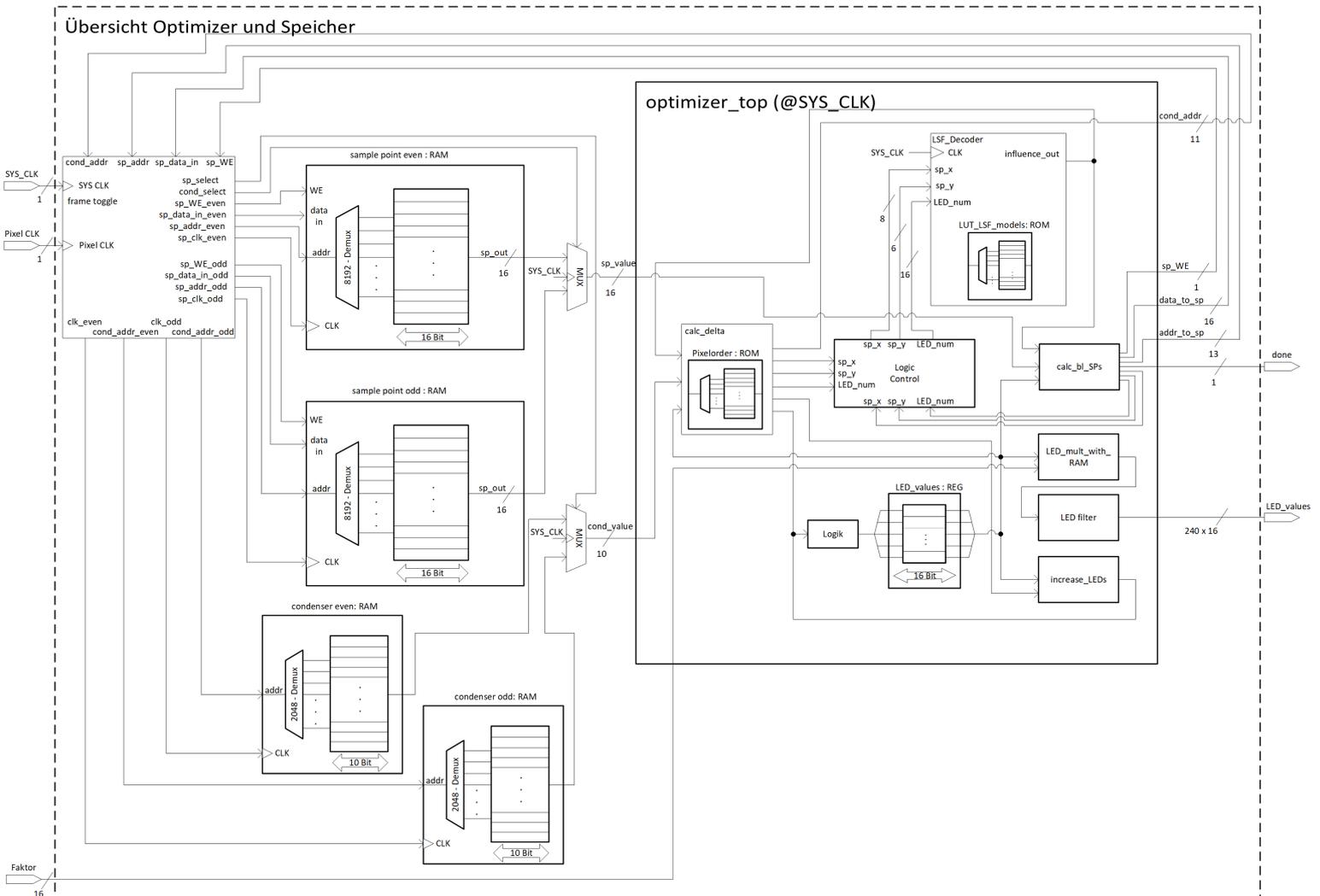


Abbildung 5.51.: (Abstrahierter) Überblick über das Modul *Optimizer\_Top* in RTL-Darstellung.

FPGA vorhandene Hardware (Blöcke), welche durch die HDL instantiiert werden. Bei der Nutzung ist aufgrund der Clock-Domain-Crossings auf mögliche metastablen Zustände der FFs zu achten [LNW10]. Im Hardware-Design kommt noch ein weiterer Domain-Wechsel im *UART*-Modul sowie beim *SPI*-Modul vor. Im *SPI*-Modul werden Register mit der Systemclock geschrieben und mit der *SPI*-Clock gelesen. Im *UART*-Modul kommen beide Richtungen vor. Insbesondere müssen an diesen Stellen Techniken angewandt werden, um das Abtasten metastabiler FFs zu vermeiden und diese in einen stabilen Zustand mit garantiert korrekten Werten überführt werden.

Das Optimizer Modul (*optimizer\_top*) selbst, wird mit der Systemclock getaktet und enthält eine hier als *Logic Control* benannte Blackbox, welche für den zeitlichen Ablauf zwischen LED-Berechnung und SP-Berechnung das Muxen der *LSF\_Decoder* Eingänge übernimmt (wird im Detail in Abschnitt 5.6.2.5 beschrieben).

Das Modul *calc\_delta* enthält das Pixelorder-ROM, welches wie in Abschnitt 5.4 erläutert, strukturiert ist. Hier wird berechnet, wie viel Lichtbeitrag pro Condensierzelle ( $i, j$ ) vorhanden ist und die Differenz ( $\Delta_{\check{L}}(i, j)$ ) zum Sollwert wie in Formel (5.15) auf Seite 161 beschrieben, gebildet. Es werden jeweils die 25 LEDs mit den höchsten Einflüssen pro Condensierzelle betrachtet. Diese LED Nummern werden zunächst in Abhängigkeit der aktuell betrachteten Zelle berechnet und in einer Datenpipeline durch den bereits vorgestellten *LSF\_Decoder* geschickt, um die entsprechenden Einflüsse aus den Modellen zu entnehmen.

Aufgrund der Betrachtung einer limitierten LED-Anzahl ( $\tilde{N}_L$ ) zum Lichtbeitrag zur Zelle ( $i, j$ ), kann der Wert 1 nicht erreicht werden (ohne Betrachtung weiterer Effekte). In Formeln (5.3) und (5.4) auf Seite 136 wird das Verhalten beschrieben. Zusätzlich jedoch, überlagert die Ungenauigkeit der Modelle bzw. der LED-Zentren wie in Abbildung 5.26 auf Seite 154 dargestellt, die maximal erreichbare Beitragssumme in einer Zelle. Somit kann der berechnete Wert durch Überlagerung der beiden Fehler auch über 100% betragen und muss während der Berechnung korrigiert und berücksichtigt werden. In Abbildung 5.52 sind die maximal möglichen Werte für eine ausgewählte BLU abgebildet. Im unteren Teil der Abbildung ist

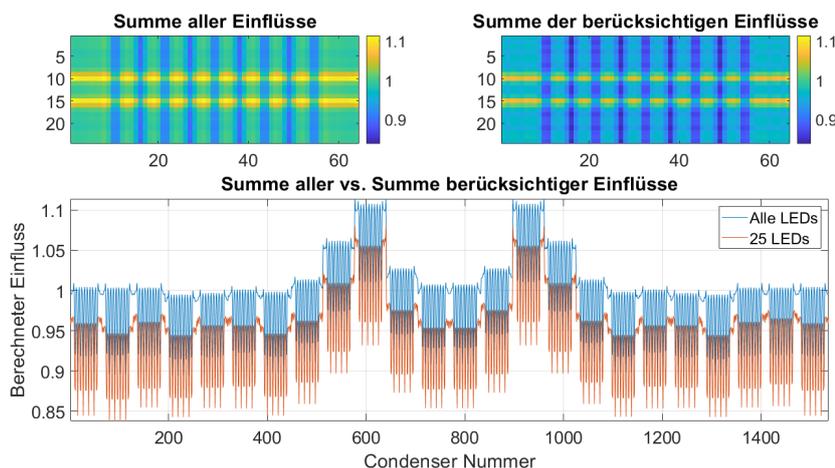


Abbildung 5.52.: Maximale Einflusssumme pro Condensierzelle ( $i, j$ ) bei Berücksichtigung der  $\tilde{N}_L = 25$  LEDs in direkter Nachbarschaft. Wie im Text beschrieben, kann der Wert pro Zelle in der Superposition  $> 1$  betragen und benötigt eine Kompensation bzw. Korrektur.

der Index linear aufgetragen und visualisiert den Unterschied zwischen der Betrachtung aller LEDs und nur der 25 umliegenden Nachbarn. Zu erkennen ist hier auch die methodeninduzierte Ungenauigkeit der LED-Positionen. Um den Fehler zu reduzieren, muss zudem Abschnitt 5.3.2 beachtet werden.

Nachdem das  $\Delta_{\check{L}}(i, j)$  berechnet wurde, werden die entsprechenden Daten, wie die potenziellen LED-Nummern, welche die LEDs repräsentieren, die erhöht werden dürfen (aus Pixelorder) an das Modul *increase\_LEDs* über eine Registerschnittstelle (*LED\_values*) weitergegeben. In diesem Modul wird entsprechend dem  $\Delta_{\check{L}}(i, j)$  die Haupt-LED erhöht, falls sie noch nicht den maximalen Wert aufweist. Für den Fall, dass diese LED bereits ihr Maximum erreicht hat, wird die nächste LED erhöht, falls dies durch die Parametereinstellung erlaubt ist. Maximal ist eine Erhöhung von 5 LEDs vorgesehen. Das neu berechnete Inkrement

der entsprechenden LED wird auf den aktuellen Wert addiert und im LED-Wert-Register gespeichert. Nach der Erhöhung wird zudem sichergestellt, dass eine Cluster-LED – wie in Abschnitt 5.5 beschrieben – falls vorhanden, auf denselben Wert angepasst wird. Ist keine Koppel-LED konfiguriert, entfällt dieser Schritt. Nachdem die LED-Werte ( $\mathbf{v}$ ) berechnet und final in LED-Werte-Register gespeichert sind, wird in Modul *calc\_bl\_SPs* das sogenannten *virtuelle* Backlight auf Stützstellenbasis berechnet und mit  $\check{\mathbf{B}}^{\text{SP}}$  repräsentiert (siehe Formel (5.20)). Dabei wird ebenfalls auf den *LSF\_Decoder*, also auf die in dieser Arbeit erstmals vorgestellten Modellierungsmethode, zugegriffen. Diese Berechnung des Backlights dient zur Pixelkompensation und wird wie in SoA, bilinear [Pre+07] in der Pixelpipeline für jedes Pixel interpoliert. Die Umsetzung der Pixelkompensation wurde in [Sch17; Jun14] bereits beschrieben. Auf die Ausführung wird an dieser Stelle verzichtet. Die Berechnung der Stützstellen unterscheidet sich jedoch fundamental von vorherigen Arbeiten, da diese Berechnung ebenfalls auf den eingeführten Modellen basiert.

$\forall (i, j) \in \{0, \dots, N_{\text{SP}}^r - 1\} \times \{0, \dots, N_{\text{SP}}^c - 1\}$  gilt:

$$\check{\mathbf{B}}^{\text{SP}} := \left( \check{b}_{i,j}^{\text{SP}} \right) = \frac{\sum_{l=0}^{N_L-1} \check{\mathbf{L}}_{(i,j),l}^{\text{SP}'} \cdot \mathbf{v}(l)}{\underbrace{\sum_{l=0}^{N_L-1} \check{\mathbf{L}}_{(i,j),l}^{\text{SP}'}}_{\text{maximaler Wert pro SP}}} \quad (5.20)$$

mit  $\check{\mathbf{L}}_l^{\text{SP}'} := \left( \check{\mathbf{L}}_{(i,j),l}^{\text{SP}'} \right)$

In obiger Formel stellt  $\check{\mathbf{L}}_l^{\text{SP}'}$  den modellbasierenden Einfluss einer LED  $l$  auf Sample-Point-Ebene dar (vgl. Seite 155). Bei der Berechnung aus Formel (5.20) wird die Korrektur der einzelnen Stützstellen durch die Normierung mit dem rechnerischen Maximalwert der Superposition der Einfluss-Modelle deutlich. Da die Berechnung des SPs sequentiell in einer Pipeline abläuft, ergibt sich für den aktuellen Prototyp folglich eine Berechnungszeit (ohne Pipelinevorlaufzeit) bei einer Pipelinefrequenz von 200 MHz mit  $N_L = 240$  und  $N_{\text{SP}} = 6321$ :

$$t_{\text{SP}} > \frac{N_L \cdot N_{\text{SP}}}{200 \cdot 10^6 \text{ s}^{-1}} = 7,58 \text{ ms}$$

Bei einer Bildwiederholfrequenz von 60 Hz entspricht diese Zeit über 45 % der Bilddauer. Für die Bestimmung der gesamten Berechnungsdauer müssen noch  $t_{\text{LEDs}} = 3 \text{ ms}$  (abhängig von der Iterationszahl und Pixelorder) für die Berechnung der LED-Werte in den weiteren Modulen im Optimizer addiert werden.

In diesem Abschnitt wurde auf die wichtigsten Module eingegangen und die Nutzung der neu entwickelten Methoden aufgezeigt. Anhand der Berechnungszeit lässt sich bereits erkennen, dass ein Großteil der Bilddauer für die Berechnung der LED-Werte und der resultierenden Sample-Points benötigt wird. Um innerhalb eines Frames eine Terminierung des Algorithmus für den Echtzeit-Betrieb deterministisch zu garantieren, muss das *Timing* der einzelnen Module untereinander sowie die geteilte Nutzung des *LSF\_Decoders* beachtet werden. Darauf wird im nächsten Abschnitt eingegangen.

### 5.6.2.5. Zeitlicher Ablauf und Übersicht

In Abbildung 5.45 auf Seite 176 wurde die Blockstruktur ohne zeitlichen Bezug als Datenflussdiagramm dargestellt. Hier wird nun der zeitliche Ablauf des Algorithmus in der Hardwareimplementierung erläutert

und veranschaulicht.

Dieser Ablauf wird in Abbildung 5.53 dargestellt. Die gezeigten Module entsprechen denjenigen aus Abbildung 5.45 und übernehmen auch dessen Farbkodierung. In Blau sind die „ROMs“ hinterlegt, während in Rot die „RAMs“ sowie Register dargestellt sind. Die einzelnen Module sind in Grün visualisiert.

Aus der Abbildung wird deutlich, wann bzw. an welcher Stelle der Prozessierung die einzelnen Module genutzt werden. Es sind 3 konsekutive Bilder zu sehen, bei denen es sich um das Bild  $i$  sowie seinen Nachfolger  $i + 1$  bzw. Vorgänger  $i - 1$  handelt. Die rote Zeitachse hat ihren Ursprung zur Zeit  $t_0$  beim ersten validen Pixel des Bildes  $i$ .

Das *VSYNC* Signal ist für eine definierte Zeit zwischen zwei Frames aktiv. Der Datenpfad der Pixel ist mit grünen Pfeilen dargestellt. Gut erkennbar ist das angedeutete *Delay* zwischen dem Eingang und dem Ausgangsbild. Es entspricht dem Versatz, welcher durch die Vorlaufzeiten der einzelnen Pixelpipelineinstufen entsteht. Insgesamt handelt es sich hierbei um nur wenige Pixelclockzyklen im niedrigen dreistelligen Nanosekundenbereich und ist für die Wahrnehmung vernachlässigbar. Es ist jedoch darauf zu achten, dass die Sync-Signale (insbesondere das *DE*) entsprechend dem Delay ebenfalls verzögert werden, um keinen Versatz im dargestellten Bild zu bekommen. Ebenfalls zu erkennen ist der Wechsel zwischen den beiden SP-RAMs und auch zwischen den Condenser-RAMs nach jedem Frame. Hier kommt unter anderem ein Clock-Muxer zum Einsatz um die RAMs mit verschiedenen Frequenzen takten zu können. Beim Condenser läuft ein RAM synchron mit der Pixelclock und speichert die aktuellen aus den Pixeln berechnete Werte, während der zweite RAM mit der Systemclock getaktet für die Berechnung der LED-Werte genutzt wird. Analog verhält sich der SP-RAM. Die Pixelkompensation nimmt also den SP-RAM-Inhalt des vorangegangenen Frames als Kompensationsfaktor zur Erhöhung der Pixelwerte (noch im physikalisch linearen Bereich, vor der Anwendung des *Degammas*). Bei einer hohen Bildwiederholrate führt dies zu keinen sichtbaren Artefakten, da die Annahme, dass zwei aufeinanderfolgende Frames sehr ähnlich sind, zutrifft.

Weiterhin ist in der Abbildung die Berechnungszeit der LED-Werte mit  $t_{LEDs}$  und die der finalen Sample-Points mit  $t_{SP}$  markiert. Die gesamte Framerate wird hier mit  $t_{frame}$  beschrieben und entspricht  $1/60$  s bei einer Ansteuerung mit einer Framerate von 60 Hz.

Anhand des zeitlichen Ablaufs wird ersichtlich, dass eine weitere Steigerung der BLU-Auflösung oder Frames per Second (FPS) die zeitliche Limitierung weiter verschärft. Dazu werden in Abschnitt 6.5 weitere Methoden vorgestellt, um trotz Limitierung noch höhere Matrix-BLU Auflösungen abdecken zu können.

## 5.7. Ergebnisse

In diesem Kapitel wurden Methoden und reale Hardwarerealisierungen vorgestellt um Direct-Lit Local-Dimming auch für hohe BLU Auflösungen unter moderaten Hardwarekosten realisieren zu können. Dabei wurde der *Crosstalk* berücksichtigt, sodass das Powersaving im Optimum liegt.

In Abbildung 5.54 ist der Prototyp mit 133 LEDs abgebildet. Ein typisches HMI-Bild wird dargestellt. Rechts daneben wird mit einem LME-eigenen Programm, welches in *Java* implementiert wurde, der Strombedarf der LEDs dargestellt. Dieser Wert wird in Echtzeit im FPGA selbst in einem eigenen Modul berechnet und über UART an einen Computer übertragen. In Abbildung 5.55 werden Fotos des in Betrieb genommenen Prototyps mit 319 LEDs dargestellt. Bild a) und b) zeigen dabei eine gedimmte und eine ungedimmte Variante des Bildes mit dem *schwarzen Hund*. Die Schwarzwertverbesserung ist deutlich sichtbar. Dabei wurde b) länger belichtet, um die Verbesserung des Schwarzwertes zu verdeutlichen (bleibt trotz längerer

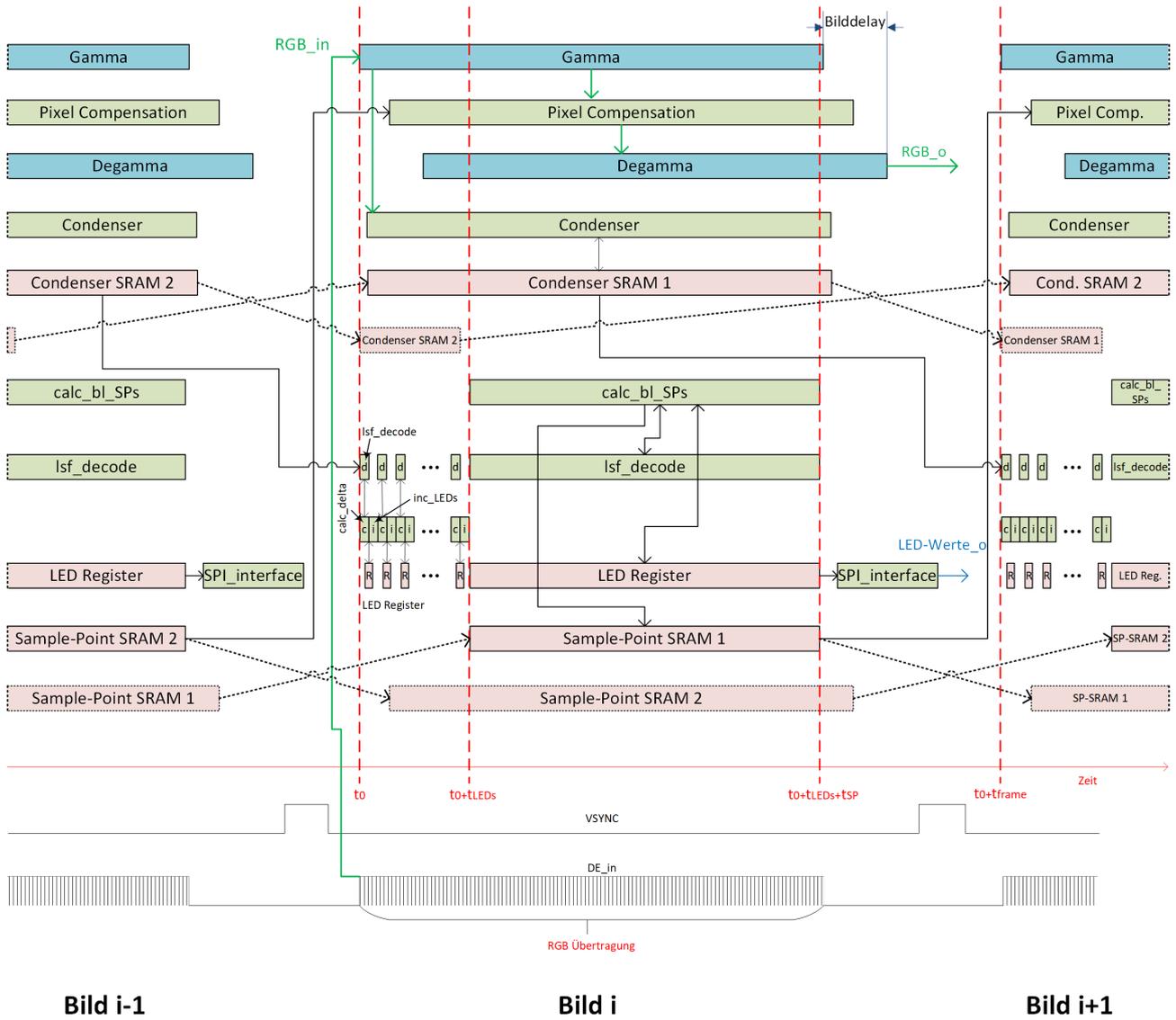


Abbildung 5.53.: Timing und zeitlicher Ablauf der FPGA-Implementierung.

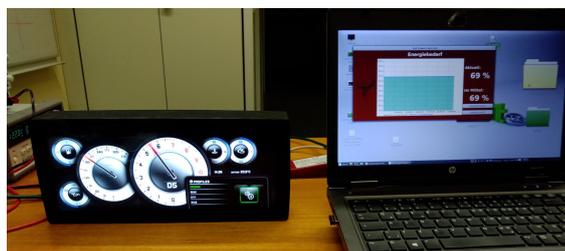


Abbildung 5.54.: Prototyp mit 133er BLU im Gehäuse (3D-Druck von Continental). Daneben Anzeige des resultierenden Strombedarfs der LEDs mit Hilfe einer GUI.

Belichtung schwarz). Die Bilder c) und d) zeigen zu zwei verschiedenen Bildern die Backlightverteilung bei 100 % LC-Transmission ( $L_{\Sigma,t}^{\text{abs}}$  mit  $t = 1$ ).



(a) All-On mit deutlich sichtbarem *Light-Leakage*. Die hellen Spots kommen durch mechanische Spannungen beim LC-Panel zustande.



(b) Mit Local-Dimming gedimmtes Bild zu a). Schwarzwert signifikant verbessert (länger belichtet als a). Powersaving = 90 %.



(c) 100 % LC-Transmission zur Anzeige der Hintergrundbeleuchtung des LME-Logos beim Dimmen. Powersaving = 68 %.



(d) 100 % LC-Transmission zur Anzeige der Hintergrundbeleuchtung eines HMI-Bildes beim Dimmen. Powersaving = 50 %.

Abbildung 5.55.: Fotos des Prototyps mit 319 LEDs. Hier wird insbesondere die Schwarzwertverbesserung deutlich.

Eine der wichtigsten Neuerungen ist die Entwicklung des LSF-Modellkonzeptes und das zugehörige Verfahren zur konsistenten Modellierung (siehe Abschnitt 5.3.2). Aus dieser neuartigen Methode entsteht der große Vorteil, dass nur noch eine geringe Anzahl an LSF-Messungen pro BLU notwendig sind, da im Vergleich zum SoA nicht mehr alle Einflüsse gespeichert werden müssen, sondern mit dem LSF-Modellkonzept abgedeckt werden. In Abbildung 5.56 wird der SRAM Bedarf des neuen Systems mit dem SoA Ansatz verglichen. Die Abszissenachse zeigt hier die Quadratwurzel der LED-Anzahl  $N_L$ . Auf der Ordinatenachse ist der benötigte SRAM für die verschiedenen Methoden aufgetragen. Der neue Modellansatz ist durchgängig gezeichnet, der SoA Ansatz hingegen gestrichelt. Da bei SoA gilt  $N_C = N_{SP}$ , wäre der Fehler bei der Berechnung des virtuellen Backlights  $\tilde{B}^{SP}$  größer als bei der neuen Methode mit unterschiedlichen Condenser- und SP-Auflösungen (siehe Abschnitt 5.2). Dies muss im Vergleich mit berücksichtigt werden. Es wird aus den Plots ersichtlich, dass der SRAM-Bedarf zwar initial, bei wenigen LEDs höher ist als beim SoA, jedoch sehr schnell einen Break-Even Punkt erreicht, da der SRAM-Bedarf, bis auf die Anpassungen der Bitbreiten bei der Adressierung konstant bleibt. Damit wird für zukünftige Local-Dimming Systeme mit einer hohen LED-Anzahl der Grundstein gelegt, um trotz der Herausforderungen, die Hardwarekosten möglichst gering halten zu können. Beispielsweise liegt der SRAM Bedarf des SoA mit 1536 Condenserezellen und  $N_L = 240$  bei  $> 6,99$  MBit während er mit dem neuen Ansatz bei 0,54 MBit liegt und somit um den Faktor  $\approx 13$  geringer ist. Davon entfallen  $1536 \cdot 152 \text{ Bit} = 233,472 \text{ kBit}$  auf die neue Pixelorder inklusive der vorgestellten Clustering Methode. Es ist zu erkennen, dass die Steigung des SRAM Bedarfs bei steigendem  $N_L$  flach sind. Somit sind die vorgestellten Methoden gut für eine hohe LED-Anzahl in der BLU gut geeignet. Von dem Projektpartner *Continental Automotive* wurden zwei HMI-Testvideos zur Verfügung gestellt, welche typische HMI-Inhalte zeigen. Im Mittel ergibt sich der Strombedarf mit der 133er BLU von 46,8 % und mit

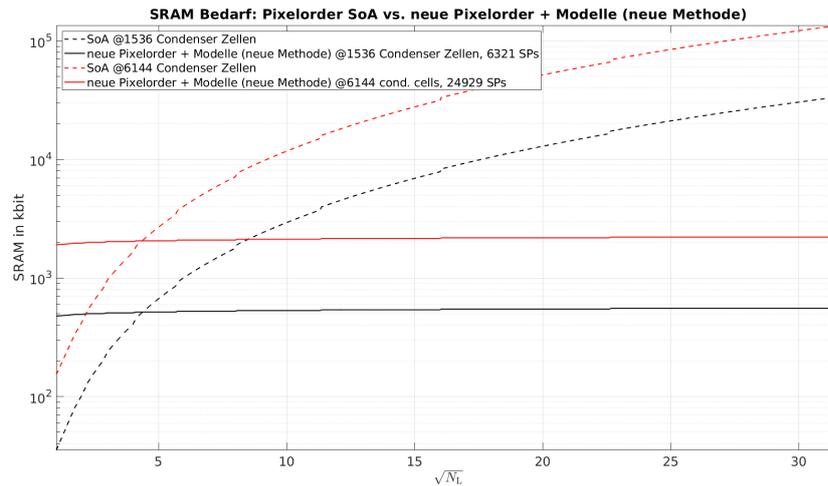


Abbildung 5.56.: SRAM-Bedarf der SoA Pixelorder im Vergleich mit der neuen Pixelorder sowie der neuen Modellmethode.

der 240er BLU von 39,9 % ausgehend vom 100 %-Bedarf des ungedimmten Bildes, also des All-Ons. Neben der Reduktion des Strombedarfs wird auch der Kontrast durch den Local-Dimming Algorithmus deutlich verbessert. Ein Beispiel ist in Abbildung 5.57 abgebildet. Es werden Eldim UMaster Leuchtdichtemessungen zu dem ungedimmten (oben) Bild und mit dem hier vorgestellten Algorithmus und einer BLU mit 319 LEDs, gedimmten Bild (unten) dargestellt. Die Leuchtdichtemessungen wurden auch hier mit der Eldim UMaster durchgeführt. Bei diesem Bild stellt sich eine Kontrastverbesserung um den Faktor 6,3 ein. Die Leistungseinsparnis beträgt 37,7 %. Der Vorteil von Local-Dimming kommt sehr gut zur Geltung. Um eine längere Betriebszeit zu erlauben und die Überlastung von sehr raren Demonstratoren zu vermeiden, wird die Leuchtdichte über die Stromeinstellung der LED Treiber reduziert. Sie kann jedoch einfach durch Erhöhung des Diodenstromes hoch-skaliert werden.

Neben den in diesem Kapitel vorgestellten neuen Methoden zur Realisierung eines Local-Dimming Algorithmus für hochauflösende BLUs, lässt sich die Qualität noch weiter steigern. Darauf wird im nächsten Kapitel eingegangen.

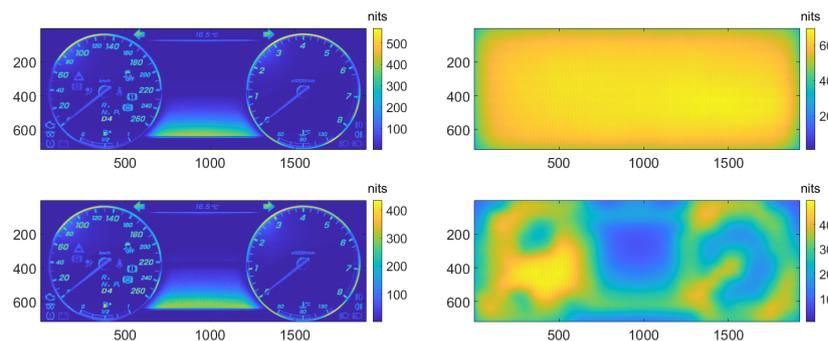


Abbildung 5.57.: Leuchtdichte eines Testbildes mit resultierendem Backlight. Oben links: ursprüngliche Leuchtdichte mit zugehörigem Backlight (oben rechts). Unten links: Leuchtdichte des gedimmten Bildes nach Kompensation mit zugehörigem Backlight (unten rechts). Entnommen aus [Xu+18b].

## 6. Weitere Methoden zur Steigerung der Bildqualität

In Kapitel 4 und 5 wurde auf Direct-Lit BLUs eingegangen und neue Methoden für die Realisierung eines Local-Dimming Algorithmus vorgestellt und implementiert. Bis hier wurde der Algorithmus auf die Performance beim Powersaving ausgelegt. Darüber hinaus wird die Bildqualität verbessert, da zugleich der Kontrast erhöht wird. Für automobiler Anwendung ist das gesamte Erscheinungsbild des Displays in Verbindung mit seiner Integration ins Armaturenbrett oder Mittelkonsole besonders wichtig. Insbesondere Premium-OEMs legen Wert darauf. Eine gelungene Integration und Bildqualität vermittelt dem Fahrer das Gefühl ein Premiumprodukt zu besitzen. Dies kann auch als *Wertigkeit* bezeichnet werden. Insbesondere bei Nachtfahrten und dunklen Umgebungen, spielt das Powersaving eine etwas geringere Rolle, da das Display keiner Sonneneinstrahlung und somit Hitze ausgesetzt ist. Zudem wird das Display bzw. dessen Hintergrundbeleuchtung stark verdunkelt, um eine Blendung zu vermeiden. Stattdessen tritt die Bildqualität in den Vordergrund. Ziel ist es, eine nahtlose Integration im Automobil zu erreichen (*seamless*). In Abbildung 6.1 ist ein Beispiel des *Postkarten*-Effektes abgebildet. Es wird ein gedimmtes LCD gezeigt, bei dem jedoch ein leichter Unterschied zwischen schwarzem Rahmen und der *Active Area* des Panels noch sichtbar ist.



Abbildung 6.1.: Postkarteneffekt auf Local-Dimming Prototyp. Zur besseren Lesbarkeit auf Papier wurde das Histogramm angepasst und der Kontrast erhöht.

Insbesondere in der beschriebenen Situation im Dunkeln fällt der Unterschied negativ auf. Die Ursache liegt in der Imperfektion des LCD wie in Abschnitt 2.4 gezeigt, welche hier durch das Light-Leakage sichtbar wird. Die Wichtigkeit für die wahrgenommene Qualität hängt also auch sehr stark von dem nativen Panelkontrast ab [HSX16b; HSX16a; CH17] und hat zudem Auswirkungen auf das *Halo-Artefakt* (Abschnitt 6.2). Typischerweise variiert das Light-Leakage zudem in der Fläche (wird auch als *Black-Mura* [Sys18] bezeichnet) und wird besonders deutlich in Abbildung 6.2 in der Messung dargestellt. Diese Qualitätsdegradation kann durch Direct-Lit Local-Dimming reduziert werden. Da die BLU und auch LC-Panel als gegeben angenommen werden müssen, kann nur algorithmenseitig versucht werden, die resultierende Leuchtdichte hinter schwarzen Flächen zu reduzieren, ohne die Inhalte in der Nähe zu verdunkeln. Gelingt dies, kann die wahrgenommene Wertigkeit des Displays erhöht werden. In diesem Fall handelt es sich meist um als *brillant* bezeichnete

wahrgenommenen Bilder, welche durch Local-Dimming lebendiger und natürlicher wirken [Grü+19; Sch+16a]. Um dieses Vorhaben zu realisieren, wird im Folgenden eine Methode vorgestellt, welche die schon vorhandenen Vorteile des Local-Dimming Systems noch weiter ausbauen kann.

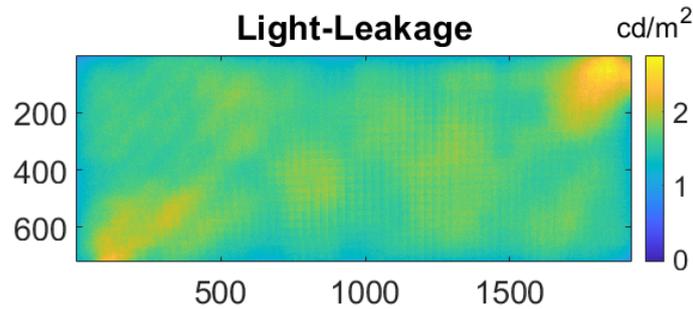


Abbildung 6.2.: Eldim UMaster Aufnahme des *Black-Mura* eines Displays.

## 6.1. Das Boosting Konzept

Das oben beschriebene Problem resultiert physikalisch aus dem Light-Leakage des LC-Panels. Bedingt durch die LSF müssen jedoch, besonders für helle Bildinhalte mehr als eine LED betrieben werden, um die ursprüngliche Leuchtdichte dieses Inhalts beim Dimmen zu erhalten, denn wie in Abschnitt 2.5.2 und Kapitel 4 beschrieben, ist der Einfluss einer LED aufgrund des Crosstalks immer deutlich  $< 100\%$ . Dadurch müssen mehrere LEDs ihren Lichtbeitrag leisten, um beispielsweise ein weißes Pixel mit ursprünglicher Leuchtdichte darzustellen. Das Konzept des *Boostings* wird in dieser Arbeit eingesetzt und hat das Ziel, die Anzahl der LEDs, welche zur Darstellung der ursprünglichen Leuchtdichte in einem Pixel benötigt werden, zu reduzieren. Dadurch wird insbesondere im Nachtbetrieb die Wertigkeit des Displays deutlich gesteigert, da das Dimming lokaler durchgeführt werden kann und dadurch das Light-Leakage in den benachbarten Bereichen verringert wird. Somit können auch die sehr lokalen Symbole und Bildhighlights kontrastreicher dargestellt werden. Bereits in [SG06] wurde ein Boosting-Verfahren basierend auf einem Hot Cathode Fluorescent Lamp (HCFL, Glühkathoden-Fluoreszenzlampe) Backlight realisiert (*Philips Aptura HCFL Scanning Backlight*). Diese BLU eignete sich aufgrund des nominalen Duty-Cycle von  $35\%$  für Boosting. Aufgrund des Hitzeproblems wurde das Boosting auf den Faktor 1,6 begrenzt und muss auch während des Betriebes zeitlich reduziert werden, da temporal versucht wird, einen Threshold des Mittelwerts nicht zu übersteigen. Dies wird in Abbildung 6.3 abgebildet und bezieht sich auf eine hellere Szene. Die Abszisse zeigt hier den zeitlichen Verlauf der Frames an. Die Ordinate stellt die Leuchtdichte in  $\text{cd m}^{-2}$  dar. Der durchgehend blaue Verlauf zeigt hier die aus dem Boosting resultierende Leuchtdichte an, welches aufgrund des gleitenden Durchschnittswertes limitiert werden muss. Beispielhaft ist ein Bereich mit Gelb markiert. Diese Methode hat also den großen Nachteil, dass zeitlich vor dem Boosten, ein Dimmen stattgefunden haben muss, um den gleitenden Mittelwert in einem zulässigen Bereich zu halten.

In [GH07] wurde ein ähnliches Verfahren vorgeschlagen, welches auf einem adaptiven Dimming mit Histogrammanalyse basiert und auch für LED BLUs geeignet ist. Diese beiden Methoden wurden darauf ausgelegt, die Leuchtdichte temporär zu erhöhen. Insbesondere nach [SG06] kann dasselbe Bild, je nach vorgegangenen Frames, eine deutlich voneinander verschiedene Leuchtdichten aufweisen. Für die automotiven Anwendungen, wäre dieser Ansatz nicht geeignet, da die Bildqualität nicht konstant gehalten werden kann.

Zudem wurde in [GH07] zwar eine Art Crosstalk-Kompensation durchgeführt, jedoch wurde das Boosting im Nachgang eingestellt, basierend auf bereits vorhandenen LED-Werten. Dadurch ergibt sich ein nicht optimales Powersaving und eine höhere Anfälligkeit für Artefakte bei der Kompensation.

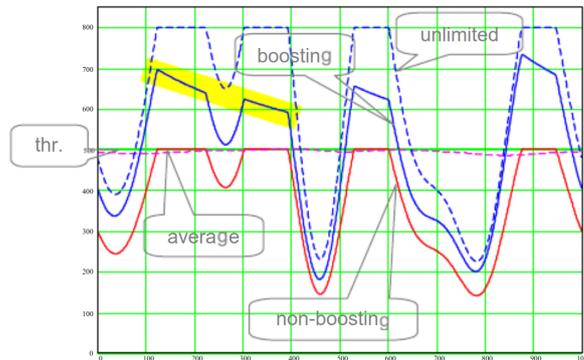


Abbildung 6.3.: Boosting in Abhängigkeit des gleitenden Mittelwertes nach Stessen und G. R. van Mourik. Magenta: Gleitender Durchschnitt. Rot: ohne Boosting. Blau: Boosting mit Limiter. Blau gestrichelt: hypothetisches Boosting ohne Limiter. Entnommen aus [SG06].

Im Gegensatz zu diesen Ansätzen, wird in dieser Arbeit ein Boosting-Verfahren vorgestellt, welches konsistent in den Berechnungsprozess für das aktuelle Frame integriert wird und effizient realisiert werden kann. Die LEDs der BLU sollen dabei weiterhin mit PWM betrieben werden. Die Stromamplitude pro LED wird im Betrieb konstant gehalten. Im nächsten Abschnitt wird im Detail auf dieses Boosting eingegangen.

### 6.1.1. Technische Umsetzung

Um die Boosting-Integration in den Local-Dimming Algorithmus besser zu verstehen, muss zunächst die Funktionsweise erläutert werden. Das Ziel ist es, weniger LEDs pro Condenserszelle für eine gegebene Leuchtdichte zu benötigen. Dies geschieht also physikalisch dadurch, dass die Stromamplitude bzw. der PWM Duty-Cycle erhöht wird. Für den Algorithmus ist es gleichbedeutend mit der Erhöhung des Einflusses um denselben Faktor. Das würde jedoch dazu führen, dass ein weiterer Multiplizierer in die Berechnungspipeline eingebunden werden müsste und somit die sowieso schon knappe Zeit weiter reduziert wird. In diesem Fall wird also jeder berechnete Einfluss mit dem Boosting Faktor multipliziert, welcher dann entsprechend bei der Berechnung der LED-Werte berücksichtigt wird. Der Duty-Cycle bzw. die PWM Auflösung kann beispielsweise auch durch die Erhöhung der vorhandenen Bits der LED-Werte erhöht werden. Somit wird für einen Boostfaktor von 2 ein Bit mehr benötigt, für einen Faktor von 4 entsprechend 2 Bit usw. für jede LED. Ist der Faktor keine Zweierpotenz, so wird der vorhandene Wertebereich nicht mehr ausgenutzt und erzeugt vermeidbaren Overhead. Jeder LED-Wert muss während der Berechnung in jeder Iteration entsprechend skaliert werden und geht auf Kosten der Berechnungszeit. Der Vorteil dieser Methode ist jedoch eine hohe Genauigkeit.

Im weiteren Verlauf kann der Begriff *LED-Wert* als Alternative zu *PWM-Wert* verwendet werden, da es sich hierbei in beiden Fällen um ein Verhältnis zum maximalen Wert handelt, jedoch in unterschiedlichen Wertebereichen, sodass sich beides auf denselben Sachverhalt bezieht und eine Unterscheidung in diesem Kontext nicht notwendig ist.

Für eine hardware-schonende Umsetzung, ergibt sich jedoch eine sinnvollere Alternative dadurch, dass

anstelle des Einflusses, der Wert der Condensierzelle um diesen Boosting Faktor reduziert wird. Ein kurzes Beispiel wird weiter unten aufgeführt. Diese Berechnung wird dann immer nur beim Fetchen der kommenden Pixelorderzeile durchgeführt und hat keinen Einfluss auf die eigentliche Berechnung des Einflusses aus dem Modell. Des Weiteren soll der Gesamtstrom innerhalb der BLU für den Boosting-Fall nicht höher sein, als der All-On Fall ohne Boosting um Überhitzung und ein zu schnelles Altern der Komponenten zu vermeiden. Als Referenz wird ein weißes Bild mit einem konstanten Diodenstrom und ein Duty-Cycle von 100 % für die LEDs angenommen. Da der Duty-Cycle bereits 1 beträgt, kann dieser nicht weiter erhöht werden. Um trotzdem ein Boosting zu realisieren, wird die Stromamplitude erhöht und somit der PWM-Wert reduziert, um die ursprüngliche Leuchtdichte zu erhalten. Selbstredend muss der maximale Diodenstrom, sowie die Spezifikation der Treiber bei der Wahl der Stromamplitude beachtet werden.

Um auch ein *globales* Abdunkeln der BLU zu implementieren (keine Kompensation), um vom hellen Tagbetrieb über Dämmerung zum dunklen Nachtbetrieb schalten zu können, wird ein globaler Faktor  $GF \leq 1$  definiert. Je nach gewünschter Auflösung der Stufen und der maximalen Leuchtdichte werden entsprechend viele Bit benötigt. Beispielsweise wird für ein  $1000 \text{ cd m}^{-2}$  Display bei gewünschter globaler Abstufung von  $16 \text{ cd m}^{-2}$  alleine 6 Bit für den globalen Anteil benötigt. Analog muss auch der maximal mögliche bzw. maximal definierte Boostingfaktor berücksichtigt werden. Für einen Faktor von maximal 8 kommen also weitere 3 Bit hinzu. Der Rest entfällt auf die weitere Verfeinerung durch Local-Dimming. Somit wird deutlich, dass eine PWM-Auflösung von 16 Bit keineswegs zu hoch angesetzt ist, sondern je nach Spezifikation auch ausgereizt werden kann. Der hier genutzte Treiber [Tex14] liefert diese Auflösung. Im Folgenden sei der physikalische Boost als  $PB$  definiert (Faktor der Stromerhöhung). Als Beispiel sei der Strom von 20 mA auf 50 mA erhöht und entspricht folglich einem  $PB = 2,5$ . Dann gilt für den *maximal möglichen* Boostfaktor ( $PF$ ):

$$PF = \frac{PB}{GF}$$

Zudem wird der *maximal erlaubte* Faktor  $MF \geq 1$  definiert, um das Boosting auf einen sinnvollen Wert begrenzen zu können. Dieser Parameter erlaubt somit auch eine Limitierung eines Boostes, welcher durch eine globale Leuchtdichtereduktion mit  $GF < 1$  im Nachtbetrieb inhärent genutzt werden kann, wie später erläutert wird. Zugleich wird verhindert, dass die Auflösung der LED-Werte nicht zu stark reduziert wird. Der tatsächliche Boost (*real Boost*) ergibt sich aus

$$RB = \min(PF, MF)$$

In dieser Arbeit wird der  $PB < 8$  definiert und für die Berechnung im FPGA als Festkommawert mit einer Quantisierung Q3.7 gespeichert. Um die gewünschte Reduzierung der Condenserwerte mit möglichst wenig Zeit (Clockperioden) und Hardware durchzuführen, wird anstelle der Division, der Condenserwert mit dem Kehrwert von  $RB$  multipliziert. An dieser Stelle reicht eine Auflösung des Kehrwertes von 10 Bit. So verringert sich der Wert einer Condensierzelle mit dem originalen Wert von 1023 auf den Wert  $\lfloor 1023 \cdot 0,4 \rfloor = 409$ . Ein weißes Bild liefert also optimalerweise für die LED einen Duty-Cycle von 40 % (bei  $PB = 2,5$ ) und liefert die ursprüngliche Leuchtdichte. Bei einem Bild mit einzelnen *Highlights* hingegen kann die aus der Boosting-Methode resultierenden PWM Reserve von  $100 \% - 40 \% = 60 \%$  ausgenutzt werden und somit auch über die nominalen 40 % steigen, um eine entsprechende Leuchtdichte zu liefern. Dies geschieht durch die Erhöhung des Einflusses bzw. die Verringerung der Zielwerte (Condenser) systeminhärent. Eine weitere Änderung betrifft nun die Berechnung der Sample-Points. Diese müssen jetzt mit dem Faktor

## 6.1. Das Boosting Konzept

$RB$  multipliziert werden, um eine korrekte Kompensation zu erhalten, analog zu Formel (6.3) auf Seite 221. Final, also vor der Übertragung der berechneten PWM-Werte per SPI an die Treiber, müssen die eigentlichen LED-Werte durch einen Skalierungsfaktor ( $Q0.16$ ) korrigiert werden, um auch den globalen Faktor, sowie den physikalischen und tatsächlichen Boost bei der Ansteuerung zu berücksichtigen. Dieser Skalierungswert entspricht dem *Scale-Faktor*  $SF \in [0, 1]$ , mit welchem die berechneten LED-Werte multipliziert werden. Für einen finalen PWM-Wert  $v_{\text{final}}(I)$  einer LED  $I$  und dem berechneten LED-Wert  $v_{\text{calc}}(I)$  ergibt sich folgender Zusammenhang

$$v_{\text{final}}(I) = v_{\text{calc}}(I) \cdot SF = v_{\text{calc}}(I) \cdot \underbrace{GF \cdot \frac{RB}{PB}}_{SF}$$

In Abbildung 6.4 wird der Zusammenhang der Parameter im Blockdiagramm dargestellt. Diese Parameter können extern berechnet und über UART an das FPGA-Design übertragen oder schon im FPGA als auswählbare Parametersets gespeichert. Auf Änderung wird einmal pro Frame in der Blankingzeit überprüft. Beispielsweise kann die Berechnung mithilfe der LME-GUI erfolgen und übertragen werden.

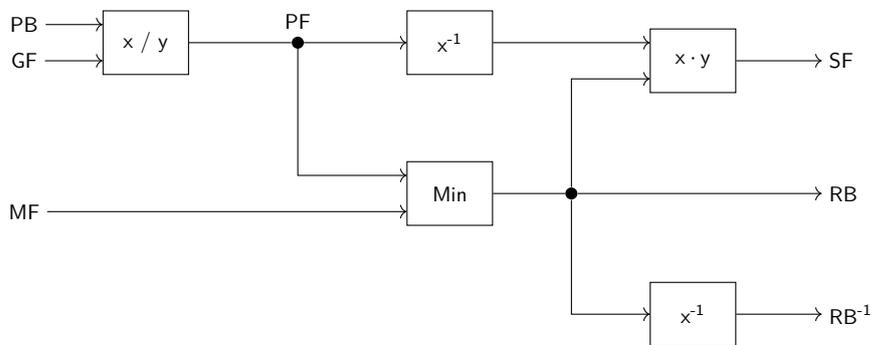


Abbildung 6.4.: Berechnung der Boostingparameter und Funktion als Blockdiagramm.

In Abbildung 6.5 werden die Auswirkungen der Parameter auf den Real Boost und auf den Duty-Cycle einer LED in Abhängigkeit des globalen Faktors  $GF$  für verschiedene maximal erlaubten Faktoren  $MF$  dargestellt. Der physikalische Boost beträgt 2,5 und entspricht der eingestellten Erhöhung des LED-Stromes. In a) wird auf der Ordinate der resultierende  $RB$  Faktor aufgetragen. Die farbigen Linien stellen dabei eine Schar des Parameters  $MF$  dar. Rot gestrichelt ist die Ordinate, an der gilt  $PB = RB$  und stellt hier die untere sinnvolle Grenze für  $MF$  dar. Wird  $MF < PB$  gewählt (orange und rot), so entspricht dies einer Verdunkelung des Backlights und ist für das Boosting Konzept nicht sinnvoll. Die Berechnung kann mit Abbildung 6.4 nachvollzogen werden. Gilt  $MF > PB$ , so kann der globale Faktor  $GF$  dazu beitragen, den  $RB$  zu erhöhen. Dies kann im Nachtbetrieb nutzbar gemacht werden. Je geringer  $GF$  ist, desto höher ist der mögliche Boost, jedoch limitiert durch  $MF$ . Der graue Bereich stellt hier den nicht erreichbaren Wertebereich dar und wird für größer werdenden physikalischen Boost immer kleiner (wachsendes  $PB$ ), ist jedoch je nach Treiber und LED-Spezifikation festzusetzen.

In Abbildung 6.6 wird anhand eines Beispiels die Auswirkungen des Boostings mithilfe von Simulation visualisiert. Ein HMI Testbild ist (oben links) mit den zugehörigen Condenserwerten dargestellt. Es handelt sich um eine BLU mit  $N_L = 240$ . Der physikalische Boost ist durch eine Stromamplitudenerhöhung von  $PB = 2$  realisiert. Der resultierende Boostingfaktor ergibt sich zu  $RB = 2$ . In der zweiten Zeile sind die

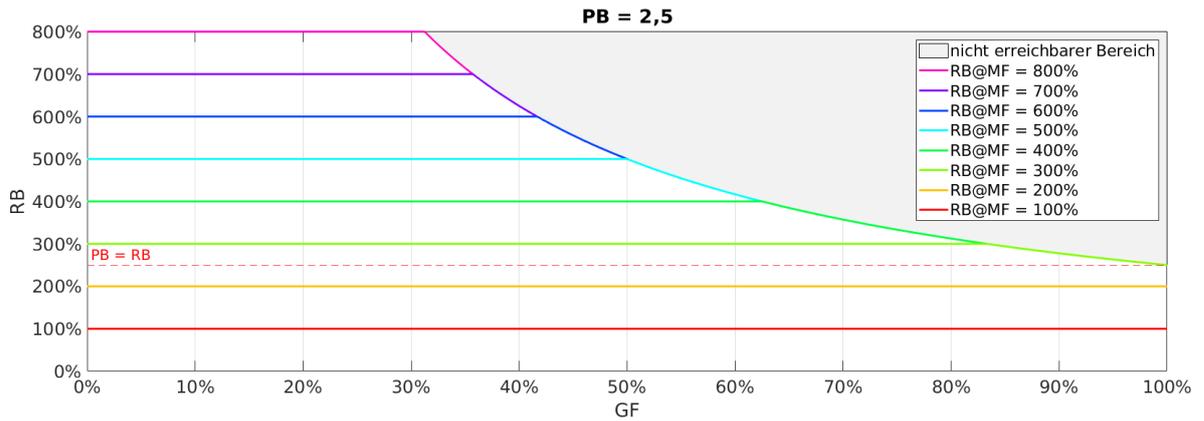


Abbildung 6.5.: Abhängigkeit des  $RB$  von dem gewählten globalen Faktor  $GF$  bei verschiedenen maximal erlaubten Faktoren  $MF$ . Physikalischer Boost (tatsächlichen Stromerhöhung des LED-Stromes) entspricht hier  $PB = 2,5$ .

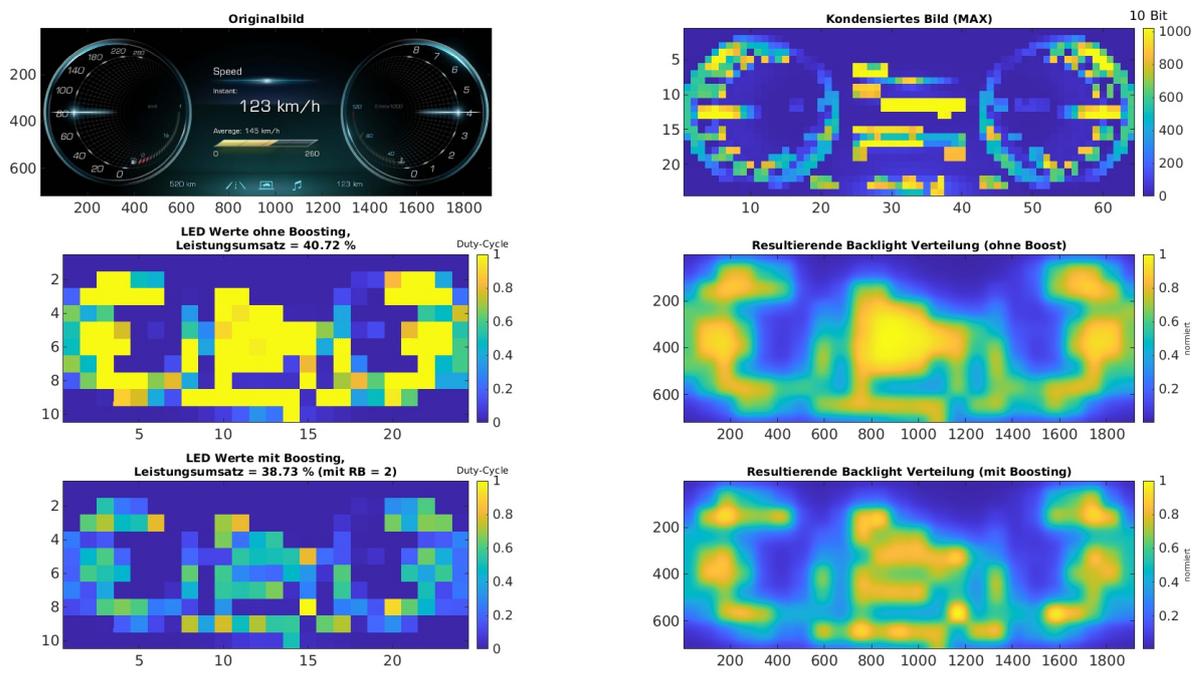


Abbildung 6.6.: Beispiel Local-Dimming ohne Boosting vs. mit Boosting Faktor  $RB = 2$ .

einzelnen LED-Werte als Duty-Cycles zu dem gedimmten, aber ungeboosteten Fall, sowie das zugehörige Backlight dargestellt. Gut erkennbar ist, dass viele LEDs denselben Wert haben, da eine LED in den meisten Fällen nicht ausreicht, um die Zielwerte zu erreichen. Im Vergleich hierzu wird im Fall des Boostings in der unteren Zeile deutlich, dass der dunkle Bereich sich weiter ausbreitet und einzelne hellere *Hotspots* durch die Berechnung entstehen (siehe auch Abschnitt 4.2.2). Die Duty-Cycles sind hier nur lokal  $> RB^{-1} = 50\%$ . Zu beachten ist die in diesem Beispiel doppelte Stromamplitude.

Der Effekt von Boosting ist hier gut erkennbar. Die Lösung ist lokaler. Dadurch erhöht sich auch der lokale Kontrast und somit auch Wertigkeit bzw. wahrgenommene Qualität des Displays. Zu erkennen ist insbesondere, dass im Tachometer in den dunklen Bereichen, sowie an den Rändern des Displays die Flächen

noch dunkler werden und somit auch den erwähnten *Postkarten-Effekt* durch einen verbesserten Übergang zum Armaturenbrett reduziert wird. In diesem Beispiel ist das absolute Powersaving fast identisch. Ist jedoch der Einfluss gering und die Anzahl der erhöhbaren LEDs begrenzt, so kann in diesem Fall durch das Boosting effektiv ein mögliches Clipping verhindert werden, da dadurch der Einfluss erhöht wird und somit auch rechnerisch weniger LEDs benötigt, um die entsprechenden durch den Condenser vorgegebenen Zielwerte zu erreichen.

### 6.1.2. Resultate

Mit der Einführung der integrativen Boosting Methode mithilfe neuer Constraints für den Local-Dimming Prozessor, berücksichtigt der Optimizer die resultierenden Reduktionen der Condenserwerte während der LED-Berechnungsphase und kann somit das Potential des Boosting in Kombination mit Direct-Lit Local-Dimming ausschöpfen. In Abbildung 6.7 werden für zwei Testbilder (in Falschfarben, erste Reihe) die berechneten, resultierenden Lichtverteilung ohne Boosting und mit Boosting  $RB = 2$  dargestellt. Zur

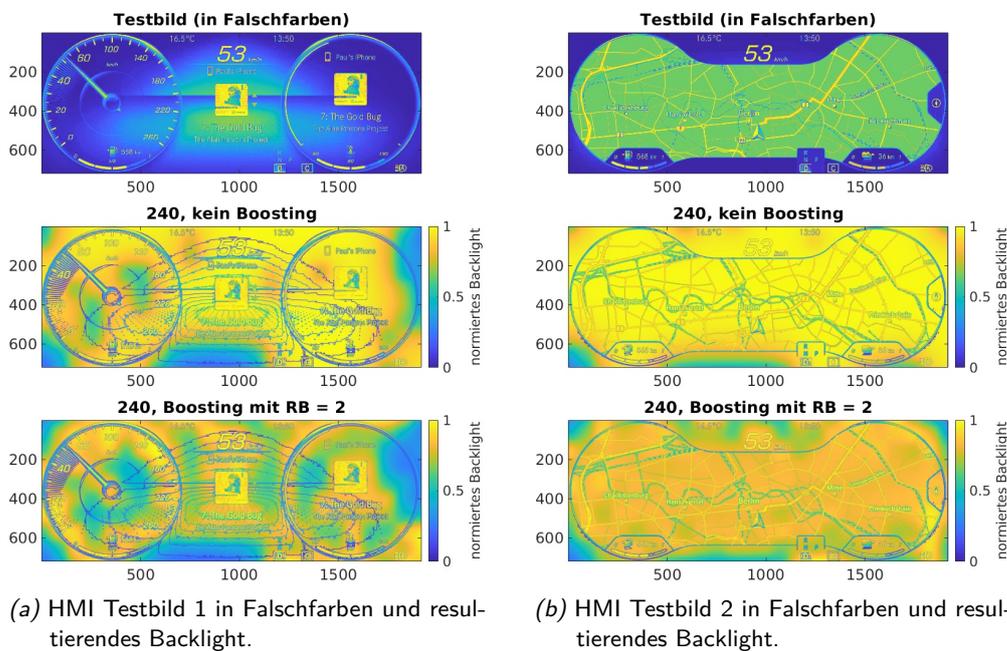
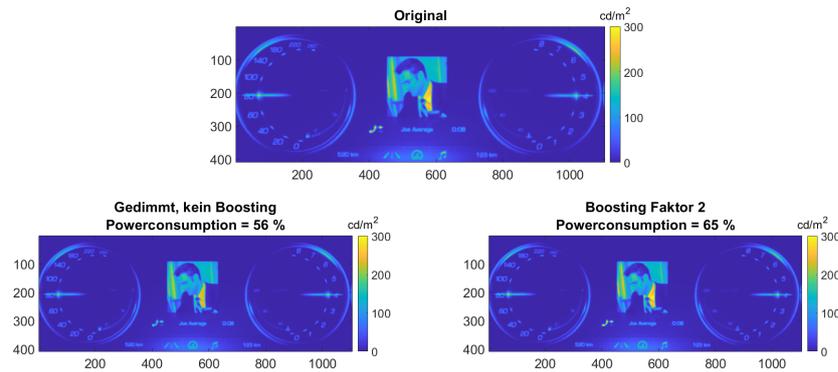


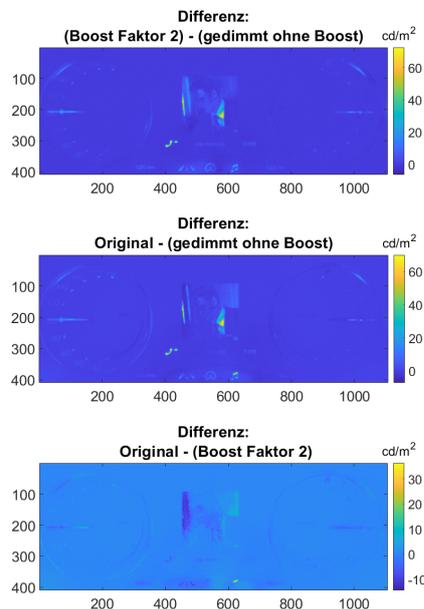
Abbildung 6.7.: Simulation zweier Testbilder mit einer 240er BLU mit  $RB = 2$  vs. ohne Boosting. Die Konturen der Testbilder sind dem Backlight überlagert.

Verdeutlich sind die Konturen der Testbilder in den Backlights darunter überlagert. Das linke Bild zeigt eine Navigationskarte und hat aufgrund der Straßen ein helleres Erscheinungsbild als das rechte Testbild. Ohne Boost ist insbesondere bei dem linken Bild, aufgrund der sich überall befindlichen hellen Stellen, der Kontrastgewinn relativ gering. Das Boosting jedoch, zeigt eine deutlich höhere Kontraststeigerung und ermöglicht eine Abdunklung der Ecken und erlaubt somit einen besseren Übergang zu der nicht leuchtenden Umgebung (Armaturenbrett), insbesondere im Nachtbetrieb. Die Verteilung ist hier deutlich lokaler. Beim rechten Bild sind bereits ohne Boosting viele schwarze Flächen vorhanden, welche durch die hier gezeigte Methode sich noch weiter ausdehnen und die Qualität des LCD-Moduls verbessern und so zur positiven Nutzererfahrung beiträgt und die Qualität des vom OEM genutzten LC-Display unterstreicht.

In Abbildung 6.8 sind Leuchtdichteaufnahmen zum Vergleich zwischen Local-Dimming mit und ohne Boosting dargestellt. Dabei wurde der Strom mit  $PB = 2$  verdoppelt und ein Boostfaktor von  $RB = 2$  gewählt. Rechts



(a) Originalbild oben. Links: Local-Dimming ohne Boosting. Rechts: Local-Dimming ohne Boosting mit entsprechendem Backlight und  $PB = RB = 2$ .



(b) Differenzen der im Titel benannten Leuchtdichteaufnahmen im Vergleich.

Abbildung 6.8.: Leuchtdichtemessung (Eldim UMaster) eines Testbildes mit und ohne Boosting sowie Differenzbildung zum Vergleich.

wird der Local-Dimming Fall mit Boosting dargestellt. Es fällt auf, dass der „Kragen“ etwas heller ist, als beim Fall ohne Boosting. Zudem ist ein leicht höherer Strombedarf bei Nutzung des Boostings zu erkennen. Dies folgt daraus, dass die Anzahl der erhöhbaren LEDs per Parameter begrenzt sind, sodass im Falle ohne Boosting etwas mehr Pixel ins Clipping laufen. Diese Begrenzung ist einstellbar, jedoch erwünscht um den lokalen Kontrast zu erhöhen und den Strombedarf weiter zu senken. Mit der Boosting Methode wird der Einfluss einer LED effektiv erhöht, sodass weniger LEDs ausreichen, um die Zielleuchtdichte zu erzeugen. Dadurch wird auch das Clipping verhindert, jedoch in diesem Fall der Leistungsumsatz etwas erhöht. Im Umkehrschluss bedeutet dies, dass ohne Boosting die eingestellten erhöhbaren LEDs nicht ausreichen, um

die Zielwerte zu erreichen.

In Teil b) sind zu den Aufnahmen die Differenzen (siehe jeweiliger Titel) der Messungen berechnet. Insbesondere die unterste Differenz zeigt, dass die Abweichungen der Soll-Leuchtdichten mit Boosting vernachlässigbar sind. Zudem ist eine Abweichung im Bereich schwarzer Pixel gewünscht, da das Light-Leakage reduziert wird. Insgesamt ist die neue Boostingmethode sinnvoll um den lokalen Kontrast weiter zu erhöhen. Damit kann auch mit suboptimalen LSFs ein besseres visuelles Ergebnis erzeugt werden.

## 6.2. Entschärfung des Halo-Artefakts

Dieser Abschnitt behandelt das sogenannte *Halo*-Artefakt, welches bei die Nutzung von Local-Dimming sichtbar werden kann [Bal+09; Ich+10; HSX16b; Tan+18]. Ein Beispiel ist in Abbildung 6.9 dargestellt. Hier ist ein *Heiligenschein*, welcher auch Halo genannt wird, um den Mond sichtbar. Es handelt sich hierbei bereits um ein mit Local-Dimming gedimmtes Bild. Das native Kontrastverhältnis des Panels ist gering. In [Sch17; Grü+16] wird zudem das Halo-Artefakt bei Edge-Lit Local-Dimming behandelt, da das Artefakt nicht nur auf Direct-Lit begrenzt ist.

Im nächsten Unterabschnitt wird das Artefakt erklärt bevor Methoden zur Entschärfung des Artefaktes vorgestellt werden.



Abbildung 6.9.: Beispiel des Halo-Artefakts bei einem Direct-Lit Gerät. Entnommen aus [Sch17].

### 6.2.1. Beschreibung des Halos

Wie bereits erwähnt, kann der Vorteil des hohen lokalen Kontrastes, welcher durch Local-Dimming eingeführt wird auch zu dem unerwünschten *Halo*-Artefakt führen. Auch hier liegt die grundsätzliche Ursache in der Unzulänglichkeit der LCDs (Light-Leakage durch den Flüssigkristall bei Transmission 0). In [Ich+10] werden zwei Typen von Halos eingeführt. Der erste Typ ist hier relevant und bezieht sich auf helle Objekte auf schwarzem Hintergrund. Ohne Local-Dimming wäre der Hintergrund vom Light-Leakage gleichmäßiger betroffen und somit der lokale Kontrast geringer. Die LED-Werte können, je nach Bildinhalt in einer bestimmten Entfernung von einem hellen Objekt sehr klein werden, während bei diesen hellen Objekten die LEDs sehr hell sind. Dadurch wird ein starker Abfall der Leuchtdichte erzeugt, welcher sichtbar sein kann. Gerade für die Automobilanwendung ist der Nachtbetrieb von besonderer Bedeutung für die wahrgenommene Bildqualität. Die mittlere Leuchtdichte wird für diesen Fall global deutlich heruntergefahren. Diese geringe, mittlere Leuchtdichte führt nach [Bar99] jedoch zu einer geringeren JND (siehe auch Abschnitt 2.2). Somit wird das Halo-Artefakt noch stärker wahrgenommen. In Abbildung 6.10 wird das sichtbare Halo-Artefakt

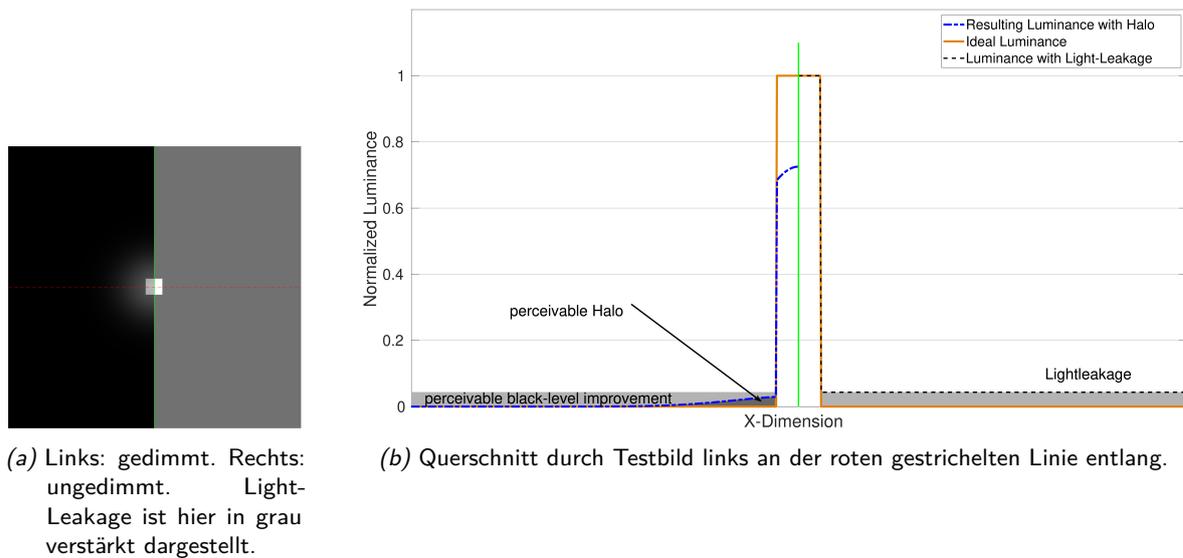


Abbildung 6.10.: Visualisierung des Halo Artefakts. Beide entnommen aus [Sch+19b].

genauer visualisiert. Für diese Simulation wurde ein Light-Leakage von 3 % gewählt. Der Local-Dimming Algorithmus durfte nur eine Haupt-LED pro Condensierzelle erhöhen. In Abbildung 6.10a ist das Beispielbild dargestellt. Der linke Teil ist das Resultat des Dimmings, während der rechte Teil das ursprüngliche, nicht gedimmte Bild darstellt (durch grüne, vertikale Linie getrennt). Das weiße Rechteck befindet sich innerhalb einer einzigen Condensierzelle. Zu Erkennen ist hier eine deutliche Verbesserung des Kontrastes, jedoch auch die Einführung des Halos. In Abbildung 6.10b ist der zugehörige Querschnitt durch die rote, gestrichelte Linie aus a) visualisiert. Die Ordinate stellt hier die normierte Leuchtdichte dar. Die orange-braune Linie zeigt den Querschnitt durch das Quadrat und beschreibt mit dem Grauwert somit die gewünschte Leuchtdichte. Durch das Light-Leakage ergibt sich jedoch für den ungedimmten Fall rechts der entsprechende Verlauf der Leuchtdichte wie dargestellt mit einer schwarz gestrichelten Linie. Ein grau-blauer Schleier ist für den Betrachter sichtbar [Sch+16a]. Wird nun eine LED direkt hinter dem Quadrat mit 100 % PWM betrieben und alle übrigen ausgeschaltet, ergibt sich der links der grünen Linie erzeugte tatsächliche Verlauf der normierten Leuchtdichte in blau gestrichelter Darstellung. Es entstehen hier zwei Bereiche. Der Bereich der wahrgenommenen Schwarzwert Verbesserung steht der Fläche dem wahrgenommenen Halo gegenüber. Also ein Qualitätsmerkmal gegen ein Artefakt. Beides induziert durch Local-Dimming und Light-Leakage. Eine Methode zur Verbesserung, ist die Erhöhung des nativen, statischen Kontrastes durch Reduktion des Light-Leakage des LC-Panels. Dies mag zwar trivial klingen, ist jedoch technologisch weitgehend ausgereizt. Nur durch weitere Verbesserung der Zellenstruktur sind noch kleinere Fortschritte zu erwarten. Aus den Versuchen in [HSX16b] geht ebenfalls hervor, dass der native Panel Kontrast sehr wichtig für (wahrgenommene) visuelle Qualität ist.

Eine weitere Methode zur Verringerung des Halos, ist die Erhöhung der LED-Anzahl. Das wahrgenommen Halo wird folglich immer geringer, das die erhöhte BLU-Auflösung die dargestellten hellen Spots im Bild immer besser trifft, sodass immer weniger Licht in die unerwünschten Bereiche hineinleuchtet. Eine sinnvolle Grenze ist erreicht, sobald ein wahrgenommenes Halo nicht mehr durch tatsächliche Lichtemission des Displays, sondern nur noch durch das Streulicht im Auge auf die Retina fällt [Ber91; MH12]. Dies bedeutet

auch, dass ein auf einem perfekten Display angezeigter heller Spot auf einer perfekt schwarzen Umgebung eine Art wahrgenommenes Halo erzeugt, welches der Physiologie des menschlichen Auges und der resultierenden Wahrnehmung geschuldet ist.

Die Erhöhung der LED-Zahl ist ebenfalls mit sehr hohen Kosten verbunden und kann für die meisten Anwendungen zum Zeitpunkt der Verfassung dieser Arbeit nicht vernünftig umgesetzt werden. In [Hsi+20] wurde zudem gezeigt, dass die LED-Anzahl sehr hoch sein muss, um die Äquivalenz mit einem OLED-Display bzw. des Kontrastes auf der Retina herzustellen. Dabei wurden Mini-LED BLUs simuliert. In Abbildung 6.11 wird die Breite eines Halos zu verschiedenen LED-Anzahlen gegenüber einer OLED-Variante aufgezeigt (linkes Bild) und auch der resultierende Verlauf auf der Netzhaut, welches mithilfe der CIE *glare* bzw. *point spread function* berechnet wurden [Hsi+20]. Daraus geht hervor, dass eine sehr hohe LED-Zahl benötigt wird, um eine OLED-äquivalente Wahrnehmung zu erzeugen. Somit ist es sinnvoll auf kostengünstige, algorithmenseitige Lösungen auszuweichen und das wahrgenommene Halo zu reduzieren. Darauf wird weiter unten eingegangen. In Abbildung 4.2 auf Seite 76 ist zudem die Abhängigkeit des Halos in einem Diagramm abgebildet.

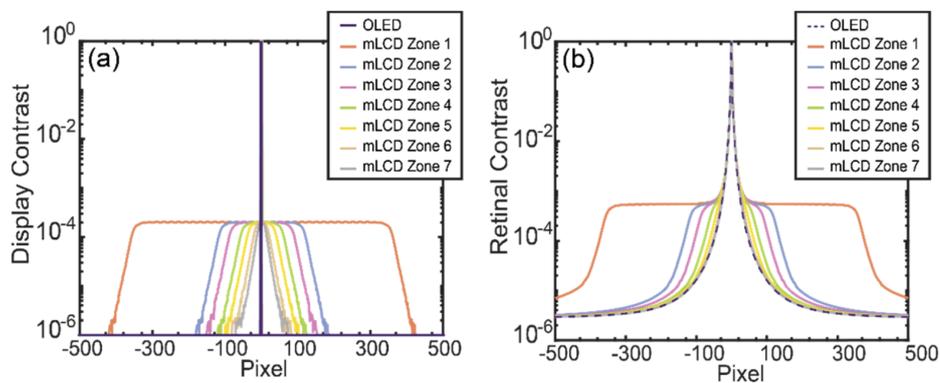


Abbildung 6.11.: Vergleich des Kontrastes zwischen einem LCD mit Mini-LED BLU und einem OLED-Display. Links: Bild auf dem Displays. Rechts: Bild auf der Retina. mLCD Zone 1-7 entspricht 18, 162, 288, 648, 1458, 5832 und 23328 LEDs in der BLU. Entnommen aus [Hsi+20].

### 6.2.2. Unterdrückung von Halo

Um das Halo-Artefakt zu unterdrücken, wird in [Zha+09] für einen LED-Wert zu bestimmen, nicht nur die direkt darunter liegende Pixelzone betrachtet, sondern auch mit der Entfernung gewichtete Zellen. In SoA ist es üblich, die Anzahl der repräsentativen Zonen gleichzusetzen mit der Anzahl der LEDs. Dabei wird nur der Bildinhalt einer Zone genutzt, um den LED-Wert abzuleiten. In dem zitierten Artikel, werden die Zonen noch etwas feiner aufgeteilt, dessen Inhalt von der aktuell betrachteten LED gewichtet in die Berechnung eingeht. In Abbildung 6.12 wird diese Aufteilung gezeigt. Der weiße Stern stellt den hellen Bildinhalt dar. Ohne Gewichtung sei der Wert für die betrachtete LED 255. Mit der dort genutzten Gewichtung ergeben sich um die LED-Position (2, 2) die folgenden 8 Bit LED-Werte.

136	208	0
208	255	88
0	8	0

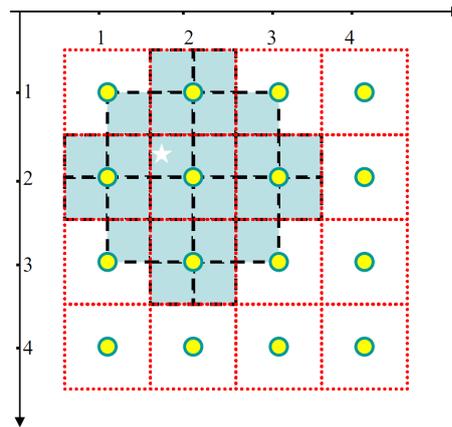


Abbildung 6.12.: Aufteilung in Zonen zur gewichteten Berechnung eines LED-Wertes. Entnommen aus [Zha+09].

Wird ein vorgeschlagener Filter auf die LED-Werte angewandt, werden die Werte wie folgt angehoben bzw. verringert:

$$\begin{array}{ccc} 128 & 128 & 128 \\ 128 & 255 & 128 \\ 128 & 128 & 128 \end{array}$$

Das Resultat ist, dass das Halo weniger sichtbar wird, da der Übergang zwischen den LED-Werten weniger steil ist. Jedoch hat diese Methode Nachteile. So wird beispielsweise der Crosstalk nicht berücksichtigt, was zu einem suboptimalen Ergebnis bzgl. Powersaving und auch Clipping führt. Zudem werden die LED-Werte durch die Anwendung eines Filters teilweise wieder verringert, was zu Fehlern bei der Berechnung eines Kompensationsfaktors führen kann. Die Idee, den Werteunterschied zweier benachbarten LEDs nicht beliebig groß werden zu lassen, um das Halo-Artefakt zu reduzieren wird hier aufgegriffen und im nächsten Abschnitt vorgestellt.

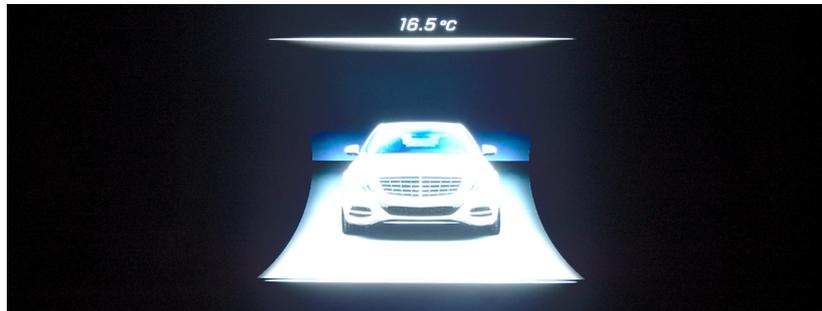
### 6.2.2.1. Räumliche Filter für LEDs

Räumliche LED-Filter wurden bereits früher, wie beispielsweise in [MLV08] vorgestellt, jedoch immer im Nachgang, also nach der Berechnung der LED-Werte angewendet. Dadurch ist im Berechnungsprozess der aktuelle LED-Wert bedeutungslos und kann nicht zur Berechnung anderer LED-Werte beitragen. Außerdem ist so eine Lösung nicht optimal. Eine andere Lösung ist die Integration des LED-Filters als Constraint in das SSC-Verfahren, mit der Bedingung, dass die LED-Werte in jeder Iteration den Filterconstraint erfüllen und somit auch durch Crosstalk-Berücksichtigung an weiteren LED-Berechnungen partizipieren. Dies liefert eine optimale Lösung, die zugleich die LED-Werte räumlich glättet. Eindimensional soll als Constraint für benachbarte LEDs  $x$  und  $x + 1$  und Faktor  $f_x$  in gelten:

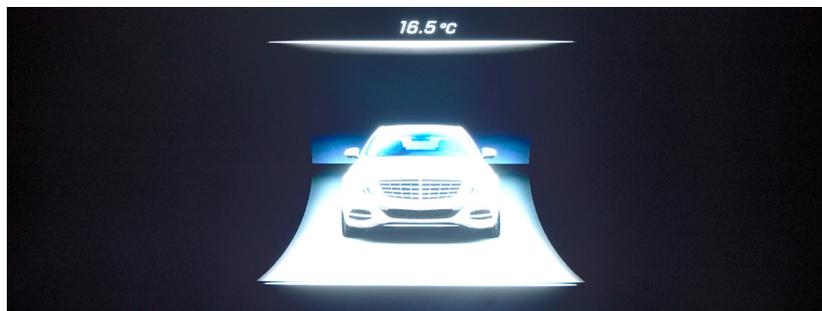
$$\frac{1}{f_x} \cdot \mathbf{v}(x+1) < \mathbf{v}(x) < f_x \cdot \mathbf{v}(x+1) \quad (6.1)$$

In [Grü+16] wurde ein Constraint wie in Formel (6.1) gezeigt, zur Realisierung eines (eindimensionalen) Filters zum Zwecke der Halo Unterdrückung auf Edge-Lit BLUs angewandt. Eine Aufnahme basierend auf

diesem Konzept wurde in [Xu+18b] veröffentlicht und ist in Abbildung 6.13 dargestellt. Es handelt sich um ein Edge-Lit mit 16 LED-Strängen an der Unterkante. Im oberen Bild ist das Halo Artefakt sichtbar, während er unten mit Anwendung des Filters als Constraint während der LED-Berechnung deutlich unterdrückt wird. Da eine Matrix-BLU zweidimensionale Strukturen besitzt, muss nun der Filter auch zweidimensional



(a) Halo-Artefakt aufgrund hohem Light-Leakage und hohem Leuchtdichtegradient zwischen den LED-Strängen sichtbar.



(b) Angepasster LED-Faktor und resultierende Reduktion des wahrgenommenen Halos.

Abbildung 6.13.: Fotografie eines HMI Bildes auf einem Edge-Lit LCD mit Überbelichtung. Entnommen aus [Xu+18b].

sein. Daher wird das Constraint um eine weitere Dimension erweitert und kann analog zu Formel (6.1) formuliert werden. Eine Visualisierung hierzu ist in Abbildung 6.14 dargestellt. Links seien die ursprünglichen LED-Werte mit einem Faktor  $f = \infty$ . Rechts sei ein Faktor  $f_{x,y} = 1,5$  gewählt und gilt für alle LEDs.

Anders als im obigen Beispiel soll der Faktor während der Optimierungsphase berücksichtigt werden. Es wird somit nach jeder Erhöhung einer LED sichergestellt, dass das Constraint erfüllt wird. Für die iterative und hardwarechonende Umsetzung wird dazu jeweils ein Faktor  $f_r$  sowie  $f_b$  definiert, welche den Faktoren aus Formel (6.1) entsprechen. Dazu ist in Abbildung 6.15 schematisch die Constraints zwischen den LEDs  $A$ ,  $B$  und  $C$  abgebildet. Dabei ist die LED  $A$  die aktuell erhöhte bzw. betrachtete LED. Die Parameter definieren jeweils den maximalen Faktor zwischen dem rechten ( $B$ ) bzw. dem darunterliegenden Nachbarn ( $C$ ). Hinzu kommt noch ein aus den beiden Faktoren abgeleiteter Faktor  $f_d$  welcher einen maximalen Unterschied zwischen dem rechten Nachbarn und dem unteren Nachbarn definiert.

Da die Berücksichtigung der Constraints nach jeder LED-Erhöhung sichergestellt werden muss, ist sie zeitkritisch. Um diese zu reduzieren, wird anstelle einer Division (auch höhere Hardwarekosten) eine Multiplikation mit den reziproken Faktoren ( $f_r^{-1}$ ,  $f_b^{-1}$ ,  $f_d^{-1}$ ) durchgeführt und in Hardware mit der Quantisierung  $Q0.5$  implementiert. Mit dieser Methode lassen sich die erlaubten Abweichungen in horizontaler und vertikaler Richtung autark vom *Upper System* wie beispielsweise über eine UART Schnittstelle einstellen und unter-

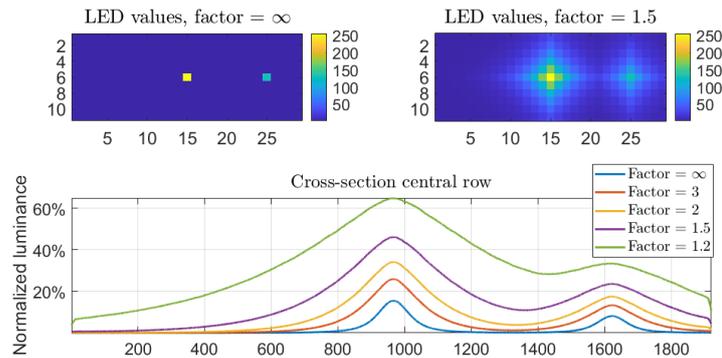


Abbildung 6.14.: Funktionsvisualisierung der LED-Faktoren bzw. des räumlichen Filters. Entnommen aus [SAX20].

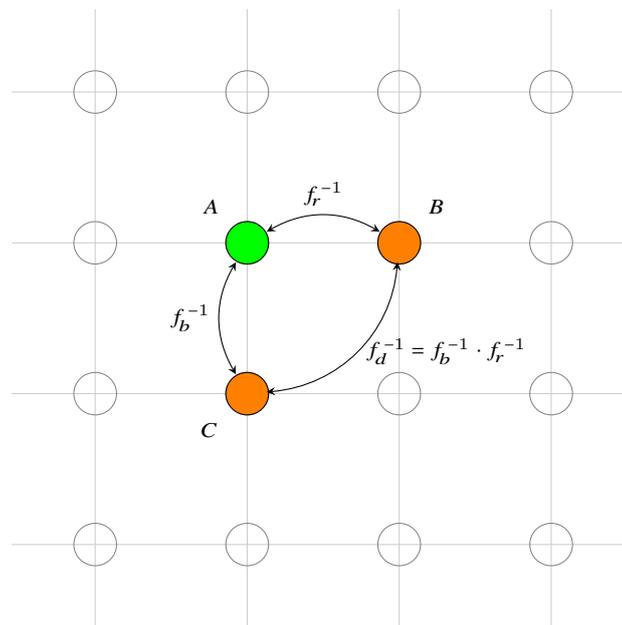


Abbildung 6.15.: An das iterative Design angepasstes Constraint. Grün stellt die aktuelle betrachtete LED dar, welche möglicherweise in diesem Schritt erhöht wurde. Orange die betrachteten Nachbarn.

schiedlich sein. Der Grund hierfür ist, dass der Verlauf in einem Bild oft naturgemäß horizontaler Inhalte aufweisen. Durch die Unabhängigkeit der beiden räumlichen Faktoren lässt sich der Algorithmus flexibler auf den entsprechenden Inhalt anpassen. Ein Beispiel der Umrechnung ist im Folgenden gezeigt. Die weitere Funktionsweise der Constraintsicherstellung wird in RTL in Abbildung 6.16 verdeutlicht.

$$f_r = 2 \Rightarrow f_r^{-1} = 0,5 \hat{=} 16_{10}^{Q0.5}$$

$$f_b = 4 \Rightarrow f_b^{-1} = 0,25 \hat{=} 8_{10}^{Q0.5}$$

$$f_d = f_r \cdot f_b = 8 \Rightarrow f_d^{-1} = f_r^{-1} \cdot f_b^{-1} = 0,125 \hat{=} 4_{10}^{Q0.5}$$

## 6.2. Entschärfung des Halo-Artefakts

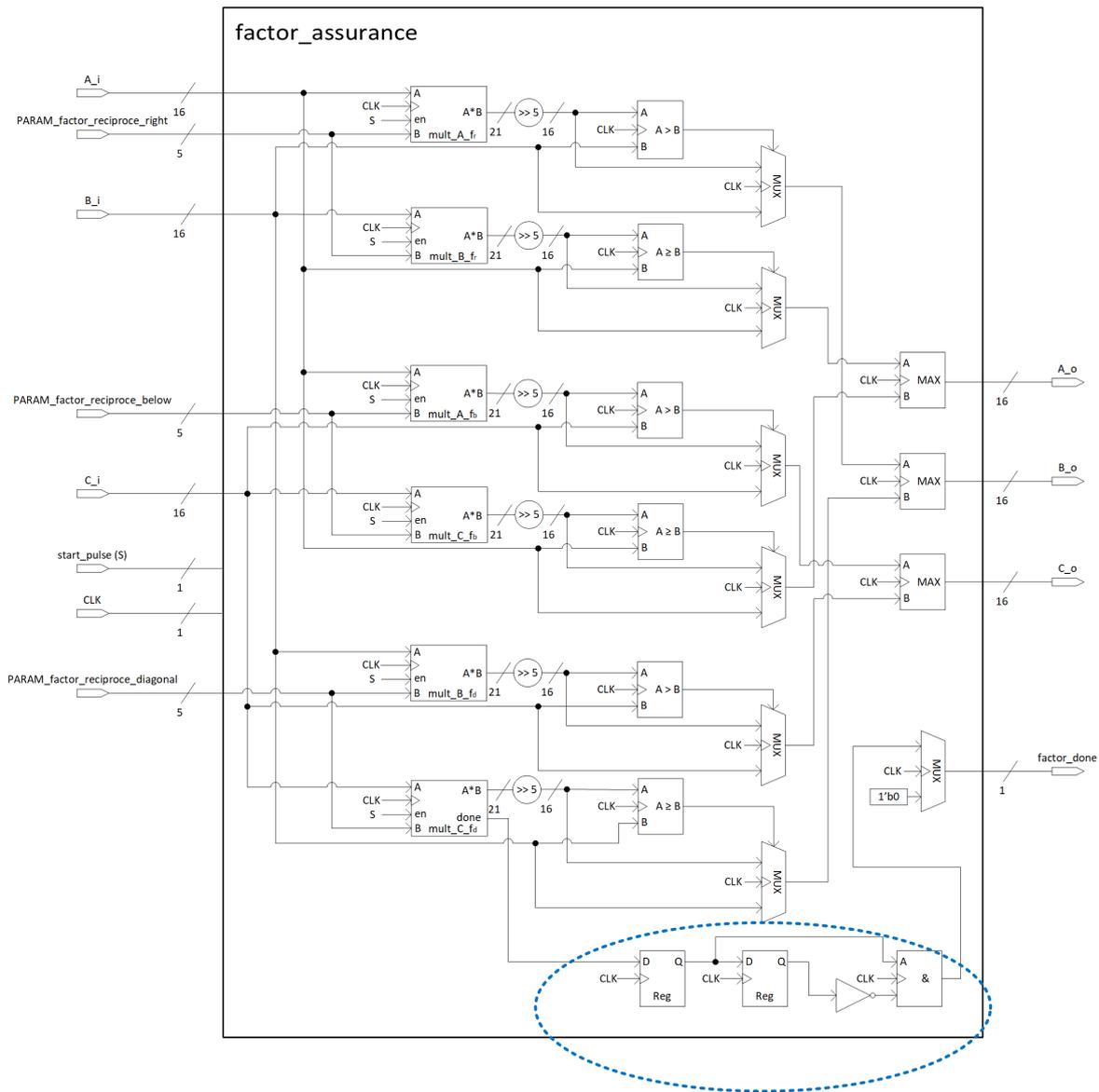


Abbildung 6.16.: RTL-Darstellung des *factor\_assurance* Moduls.

Die Signale `PARAM_factor*` stellen diese eingeführten reziproken Faktoren dar, während `A_i`, `B_i`, `C_i` den entsprechenden LED-Werten entsprechen. Die Schiebeoperation ( $\gg$ ), welche in Hardware beispielsweise durch Weglassen der entsprechenden Bits realisiert werden kann, ist notwendig um die Resultate nach der Multiplikation eines LED-Wertes (16 Bit) mit einem Faktor (5 Bit) auf die ursprüngliche Bitbreite zu reduzieren. Die mit gestricheltem Blau umrandete Logik dient hier dazu ein `done` Signal eines der parallel arbeitenden Multiplizierers entsprechend des Delays der nachrangigen Logik zu verzögern und eine Flanke am Ausgang `factor_done` zu erzeugen. Daraufhin werden die gefilterten LED-Werte wieder in das globale LED-Register geschrieben und stehen zur Berechnung des folgenden Pixelordereintrages zur Verfügung. Ein Beispiel wie sich die berechneten LED-Werte in Abhängigkeit von den eingeführten Constraints verhalten wird in Abbildung 6.17 anhand des Testbildes wie in Abbildung 6.13 gezeigt. Die Faktoren werden mit  $f_r = f_b$  von recht klein bis  $f_* = \infty$  variiert. Bei kleinen Faktoren ist die Lösung sehr global und die *abrupte*

Änderungen des Backlights wird vermieden, was insbesondere bei sehr lokalen LSFs vorkommen kann. Ein größerer Faktor erlaubt eine immer höhere Abweichung bis hin zur völligen Loslösung des Constraints. In Abbildung 6.17b sind die entsprechenden LED-Werte (hier in 12 Bit Auflösung) visualisiert und das Powersaving dazu eingetragen. Bei immer größerem Faktor nähert sich das Powersaving dem Constraint  $f_* = \infty$  an. Das Backlight wird immer lokaler und der Kontrast höher. Wie in Abbildung 6.13 gezeigt, kann jedoch ein kleinerer Faktor dazu genutzt werden, den Halo-Effekt zu entschärfen, da hohe Backlight Kontraste durch die globalere Lösung vermieden werden und ein sanfter Profilverlauf hergestellt wird.

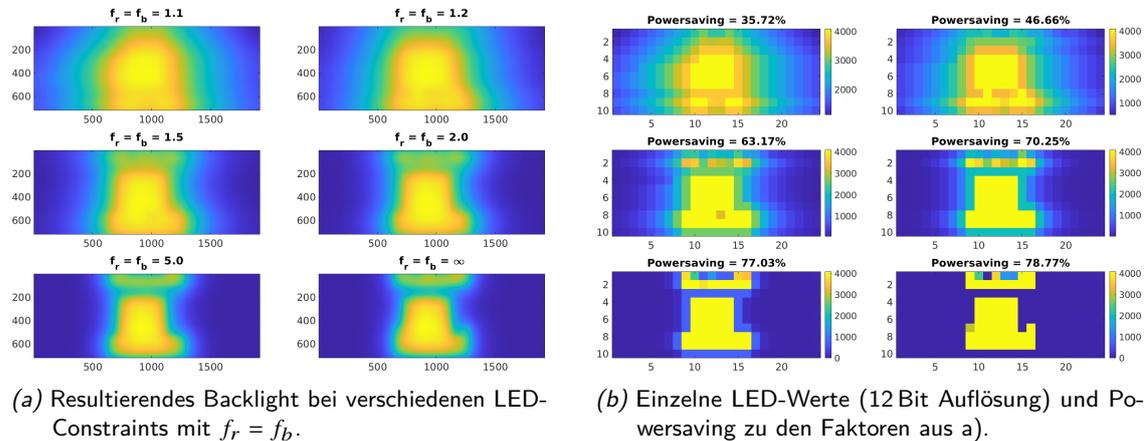


Abbildung 6.17.: Beispiel zur Auswirkung der erlaubten Abweichung zwischen LED-Nachbarn zum Testbild aus Abbildung 6.13.

Gut erkennbar ist auch, dass LED-Werte mit kleinerem Faktor stellenweise kleiner werden, als die selbigen Positionen bei hohem Faktor. Dies rührt aus der in jedem Optimierungsschritt eingehaltene Constraint und die daraus resultierenden zusätzlichen Lichtbeiträge, welche aufgrund der Faktoren zustande kommen und im Crosstalk berücksichtigt werden. Das Powersaving wird dabei geringer.

Insgesamt können diese Constraints zur Adaption der Algorithmus an die Panelqualität (im Sinne von Light-Leakage) und LSF-Form durchgeführt werden und so die Qualität eines Displays weiter gesteigert werden. Dabei muss immer ein Trade-Off zwischen sichtbarem Halo, Schwarzwert und Kontrast, sowie dem Powersaving gefunden werden.

### 6.2.2.2. LED-Min Werte

Bereits in [Bal+09] wurde gezeigt, dass das Halo Artefakt in seiner Ausprägung in unterschiedlicher Stärke auftreten kann. Insbesondere ist die Abhängigkeit vom nativen Panel Kontrast vorhanden. Es wird ein Faktor  $A$  definiert, welcher das Verhältnis der gemessenen Halo Leuchtdichte zu einem *flachen*, also ungedimmten Backlight ([Bal+09]) darstellt. Je höher der Panelkontrast ist, desto geringer wird dieser Faktor  $A$  und die *Schwere* des Halos sinkt. Ab  $A = 0,09$  jedoch sinkt die empfundene Halostärke wieder und ist schließlich bei  $\lim_{A \rightarrow 1} A = 0$ .

Basierend auf diesen Erkenntnissen wird eine weitere Methode eingeführt, um das wahrgenommene Halo bei hohem Light-Leakage zu reduzieren. Diese hat das Ziel eine geringe Grundhelligkeit der LEDs zu garantieren, um einen zu steilen Abfall der Leuchtdichte zu vermeiden. Anders als in Abschnitt 6.2.2.1 wird jedoch keine Nachbarschaftsbeziehung zwischen den LEDs garantiert, sondern ein Minimalwert  $v_{\min}$  definiert. Somit gilt

für die Initial- und finalen Werte der LEDs unter Berücksichtigung der Minimalwerte

$$0 \leq v_{\min} \leq \underbrace{\text{init}(l)}_{\text{enthält LED Initialwerte}} \leq v(l), \forall l \in \{0, \dots, N_L - 1\}$$

falls bildabhängige Initialwerte genutzt werden. Ansonsten gilt  $v_{\min} \leq v(l)$ . Somit können die LED-Werte niemals kleiner werden als das definierte Minimum. Es muss hier an dieser Stelle ebenfalls das Boosting berücksichtigt werden, sodass der Minimalwert von  $RB$  abhängig gewählt werden muss ( $v_{\min}(RB)$ ). In [Sch17] wurde solch ein Minimalwert für Edge-Lit Local-Dimming eingeführt, um die durch den Automatik-Condenser [Alb+10] eingeführten Inhomogenitäten bei dunklen, homogenen Flächen zu korrigieren. Im Falle der Automatik wurde der Condenserwert zwischen dem Mittel- und dem Maximalwert der Pixel gewählt, um das Powersaving zu erhöhen. Für automobile Anwendung muss jedoch der weiterhin der Maximalwertcondenser genutzt werden, um Clipping ausschließen zu können.

Für den Direct-Lit Algorithmus wird  $v_{\min}$  zur Reduktion des sichtbaren Halos eingeführt und lässt sich optional mit oder ohne dem LED-Faktor-Constraints aus vorherigem Abschnitt nutzen. Der Nachteil des Minimalwertes gegenüber der LED-Constraints jedoch wird insbesondere bei sehr brillanten Bildern mit großen schwarzen Flächen und dunklen Rändern deutlich, da mit dem Minimalwert der Local-Dimming Vorteil der hohen Lokalität verringert wird. Die sehr dunklen und hochwertig wirkende Randbereiche lassen in diesem Fall etwas mehr Licht durch. Für diese Fälle eignet sich der räumliche Filter mehr. Eine Kombination mit einem Minimalwert ist ebenfalls möglich und kann vom Panel und seinen Eigenschaften abhängig gewählt werden. Der Minimalwert wird, wie alle Änderungen der LED-Werte während der Berechnungsphase berücksichtigt und nicht erst im Nachgang erhöht. Somit wird ein weiterhin das Optimierungsproblem unter Berücksichtigung aller Constraints und Crosstalk sehr nahe am Optimum gelöst. In Abbildung 6.18 wird ein Beispiel, wieder basierend auf dem selbigen Testbild (vgl. Abbildung 6.17), gezeigt. In diesem Fall wird  $v_{\min}$  variiert und stellt die resultierenden Auswirkungen auf die finalen LED-Werte dar. Der Faktor-Constraint von oben wird hier zu  $f_* = \infty$  gesetzt bzw. deaktiviert. In a) wird das resultierende Backlight, basierend auf den definierten LED-Min Werten  $v_{\min}$  dargestellt. Abbildung 6.18b hingegen visualisiert die einzelnen finalen LED-Werte, zu den  $v_{\min}$  analog zu a) sowie die Reduktion des Strombedarfs. Der Strombedarf sinkt, je kleiner  $v_{\min}$  ist. Anhand dieses Beispiels wird deutlich, dass insbesondere für  $v_{\min} \geq 12,5\%$  die Lösung die Leuchtdichte hinter dunklen Flächen deutlich heller ist als bei sehr kleinen  $v_{\min}$ . Darunter leidet das Powersaving, bietet jedoch eine höhere Robustheit gegenüber dem Halo Artefakt, da die Kontrastübergänge nicht mehr so steil werden (Peak zu Background). Zudem wird aus dem Beispiel ersichtlich, dass der definierte LED-Min-Wert die Berechnung der LED-Werte während der Optimierungsphase beeinflusst, da der Crosstalk der LEDs mit berücksichtigt wird und somit bereits eine Grundhelligkeit vorgegeben wird. Die resultierenden LED-Werte können also je nach  $v_{\min}$  abweichen. Als Beispiel sind in Abbildung 6.19 die Querschnitte durch das resultierende Backlight aus Abbildung 6.18a dargestellt, sowie die LED-Werte der ersten LED-Zeile. Der Querschnitt durch das Backlight erfolgt durch die Bildzeile 15, welches im HMI-Testbild schwarz ist (oberhalb der Außentemperaturanzeige). Sehr anschaulich kann anhand des Extrembeispiels  $v_{\min} = 25\%$  dargestellt werden, dass die Mindestwerte bereits einen sehr hohen Beitrag leisten, sodass die LEDs welche auch den Bildinhalt darunter beeinflussen bereits bei kleineren Werten zur Lösung beitragen, da eine Grundhelligkeit gegeben ist. Dadurch wird das Ergebnis optimal in Bezug auf das vorgegebene Constraint.

Insgesamt sind die hier vorgestellten Constraints unter bestimmten Gegebenheiten sinnvoll. So trägt ein hoher nativer Panelkontrast, mit geringem Light-Leakage dazu bei, dass das Halo-Artefakt deutlich weniger wahrgenommen wird und sich immer weiter dem physiologisch gegebenem Streulicht des menschlichen Auges

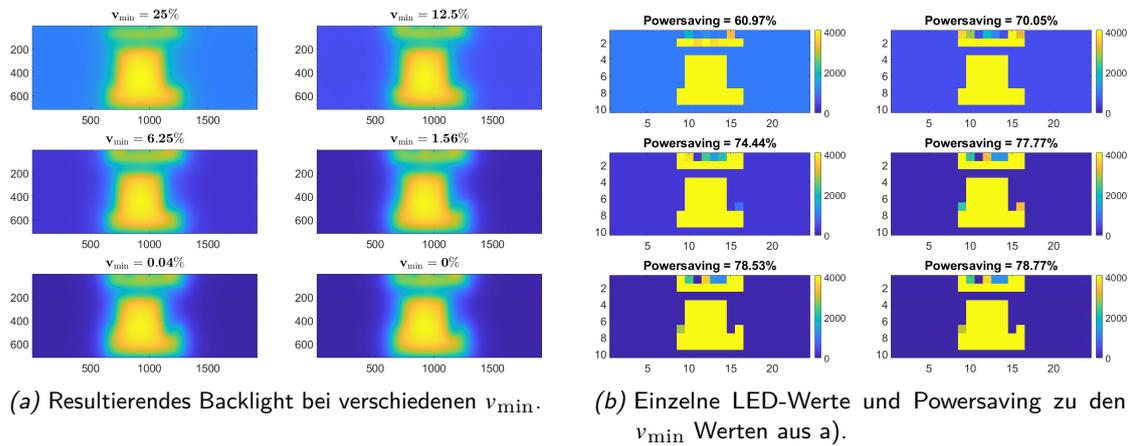


Abbildung 6.18.: Beispiel zur Auswirkung von  $v_{\min}$  auf das Backlight und LED-Lösung zum Testbild aus Abbildung 6.13.

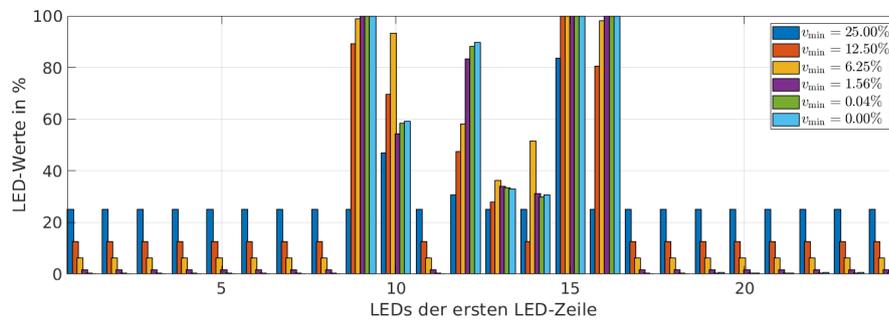
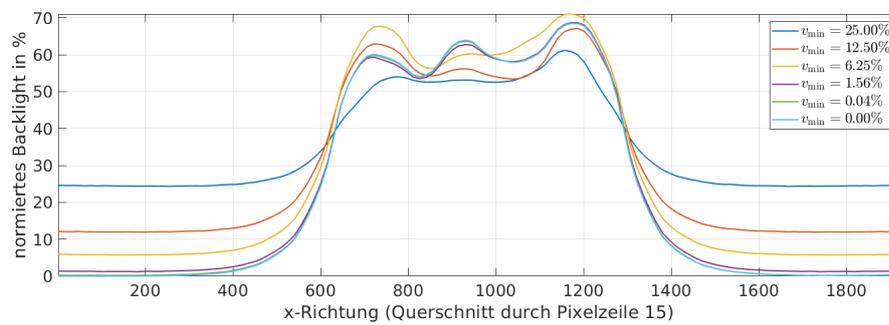


Abbildung 6.19.: Querschnitt durch resultierendes Backlight sowie Vergleich der Lösungen der ersten LED-Zeile zu zum Testbild aus Abbildung 6.13.

annähert. Ist das Light-Leakage hingegen hoch, so ist es sinnvoll die Constraints über die LED-Faktoren zu nutzen oder den minimalen LED-Wert. Es können auch beide in Kombination genutzt werden. Hier steht die Reduktion des wahrgenommenen Halos dem Powersaving gegenüber. Zudem hat das HMI-Design Einfluss auf das sichtbare Halo. Sind beispielsweise die Symbole deutlich kleiner als die LSF, so kann bei hohem Light-Leakage das Halo entsprechend groß werden und die wahrgenommene Qualität negativ beeinflussen. Die hier vorgestellten Methoden können also das sichtbare Halo reduzieren. Ein optimiertes HMI und BLU Design unterstützen die Qualitätssteigerung weiter.

Neben dem Halo spielt der Schwarzwert, insbesondere in den Displayecken eine wichtige Rolle, um die wahrgenommene Qualität weiter zu steigern. Im nächsten Abschnitt werden dazu neue Methoden vorgestellt.

## 6.3. Algorithmenseitige Ansätze zur Verbesserung des Schwarzwertes in den Displayecken im Automobil

Aufgrund des Light-Leakage beträgt die dunkelste darstellbare Leuchtdichte  $> 0 \text{ cd m}^{-2}$ . Zudem ist das Light-Leakage ortsabhängig. Da im Fall der Anwendung im Automobil eine nahtlose Integration (vgl. auch Abbildung 5.55 auf Seite 191) wichtig für die wahrgenommene Wertigkeit des Displays ist und folglich auch von den OEMs gewünscht wird, hat der Schwarzwert an den Displayecken einen sehr hohen Stellenwert. Als weitere Annahme gilt, dass die Hardware, also die BLU sowie LC-Panel vorgegeben ist und nicht optimiert werden kann. Somit bleiben noch die Methoden zur Verbesserung des Schwarzwertes in den Displayecken durch den Local-Dimming Algorithmus und den resultierenden Lösungen der LED-Werte. Diese algorithmenseitige Methoden hierzu werden im Folgenden vorgestellt.

### 6.3.1. Anpassung der Scanning Sequenz der Pixelorder

Im Allgemeinen gilt, dass die LEDs, welche sich näher am Rand bzw. an den Ecken befinden einen höheren prozentualen Lichtbeitrag auf die umliegenden Pixel liefern. Das kommt daher, dass diese LEDs weniger Nachbarn haben (durch die Displaygrenze) und somit einen höheren Einfluss auf die entsprechenden Condenserezellen aufweisen. In Abbildung 4.11 auf Seite 83 wurde das Verhalten deutlich. Entsprechend wurde auch in SoA die Pixelorderreihenfolge gewählt wie in Abschnitt 5.4.2 gezeigt wurde. Für die weitere Optimierung wurde in von Schäfer [Sch17] die Sortierung nach Haupteinflüsse dahingehend erweitert, dass zudem jede LED innerhalb der ersten  $N_L$  Pixelorderzeilen genau einmal betrachtet wurde.

Um den Schwarzwert an den Rändern zu verbessern, wird die Pixelorder nach einer neuen Methode sortiert und somit die Abarbeitungsreihenfolge der Condenserezellen und den zugehörigen LEDs wie folgt angepasst. Dabei seien die Pixelorderadressen als  $\mathbf{addr}(i), \forall i \in \{0, \dots, N_C - 1\}$  beschrieben wie bereits in Abschnitt 5.4.2. Die Reihenfolge der Condenserezellen in  $\mathbf{addr}$  sollen folgenden Bedingungen absteigend genügen

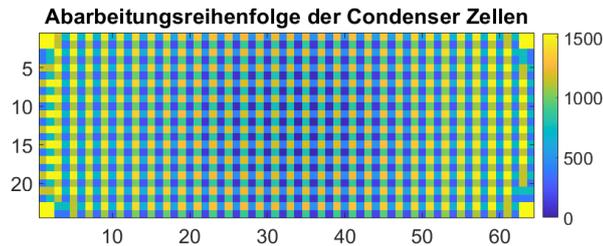
1. ( $B_1$ ): Condenserezellen in den Ecken werden jeweils am Ende einer Pixelorderiteration betrachtet. Somit werden auch die zugehörigen Haupt-LEDs an das Ende einer Iteration verschoben. Somit gilt für die Menge der an den Rändern liegenden Zellen  $C_c$ 

$$C_c = \left\{ C_p \mid \text{LED}_{p,l}^M \text{ mit } l \in \{0, N_L^c - 1, N_L^c \cdot (N_L^r - 1), N_L - 1\} \right\}$$
dann gilt für die letzten  $n$  Einträge der Pixelorder
$$\mathbf{addr}(n) \rightarrow C_c \text{ mit } n \in \{N_C - |C_c|, \dots, N_C - 1\}$$
2. ( $B_2$ ): Die LEDs, welche nicht an den Displayecken liegen werden absteigend nach Abstand zu den Ecken sortiert, sodass die LEDs in der Mitte des Displays in der Reihenfolge priorisiert werden. Die erste betrachtete Condenserezelle wird also von der Haupt-LED am meisten beeinflusst, welche von den Ecken am weitesten entfernt liegt.
3. ( $B_3$ ): Condenserezellen werden nach Abstand von der Haupt-LED sortiert.

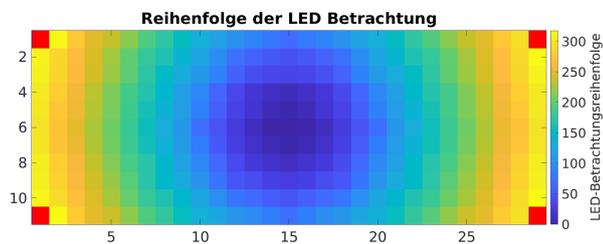
Insgesamt gilt also für die Sortierung der Condenseradressen  $\exists \mathbf{addr} : B_1 \wedge B_2 \wedge B_3$ , sodass alle Bedingungen erfüllt werden. Diese Sortierung führt dazu, dass die Wahrscheinlichkeit, dass die Eck-LEDs möglichst

dunkel sind steigt, da diese am Ende der Pixelorder betrachtet werden und vom bereits vorhandenem Crosstalk profitieren können. Somit wird zwar ein etwas schlechteres Powersaving Ergebnis aber ein besserer Schwarzwert in den Ecken erwartet, da nicht mehr die optimalste LED betrachtet wird. Der Vorteil solch einer Sortierung kann jedoch nur ausgespielt werden, wenn die Iterationsstufen grob und Iterationsanzahl gering ausfallen. Dies trifft insbesondere auf hochauflösende BLUs zu, da hier der zeitliche Aspekt ausschlaggebend ist.

Wird diese neue Sortierungsmethode auf die 319 BLU angewandt, so ergibt sich eine Abarbeitungsreihenfolge der Condenserezellen (durch die Pixelorder definiert) wie in Abbildung 6.20 gezeigt. Die Reihenfolge startet



(a) Neue Condenserezellenreihenfolge in der Pixelorder. Gelb entspricht den letzten und Blau den ersten Einträgen.



(b) LED-Positionen und farblich dargestellte Reihenfolge innerhalb der ersten  $N_L$  Einträge der Pixelorder von Blau zu Gelb. Erkennbar mit rot markiert sind die Eck-LEDs. Diese werden *nicht* betrachtet. Sie sind am Ende der Pixelorder zu finden.

Abbildung 6.20.: Resultierende Abarbeitungsreihenfolgen der Condenserezellen und die daraus implizierte Reihenfolge der LEDs, welche in den ersten  $N_L$  Pixelordereinträgen als Haupt-LEDs betrachtet werden.

bei 0 (blau) und steigt bis 1535 (gelb) an. In diesem Fall sind  $2 \cdot 2$  Condenserezellen jeweils die Eckzellen, welche von den 4 Eck-LEDs den Hauptlichtbeitrag erfahren. Im unteren Teil (c) werden die ersten  $N_L$  Haupt-LEDs Einträge aus der Pixelorder dargestellt. Dabei wird in einer  $N_L^r \times N_L^c$  farblich dargestellt, an welcher Stelle der Pixelorder die entsprechende LED (durch Zeilen und Spalten beschrieben) betrachtet wird. Mit rot markiert sind hier die LEDs, welche innerhalb der ersten  $N_L$  Pixelordereinträge *nicht* auftreten. Diese entsprechen den Eck-LEDs, da sie erst am Ende der Pixelorder eingetragen sind. Zudem wird sichtbar, dass die LEDs-Betrachtungen in der Displaymitte starten und dann Richtung Ränder laufen.

Mit dieser Methode wird zudem inhärent sichergestellt, dass die Condenserezellen so angeordnet werden, dass jede einzelne LED, außer der Eck-LEDs (hier also  $319 - 4 = 315$ ) erst wieder betrachtet wird, wenn alle übrigen bereits an der Reihe waren. Dies entspricht bis auf die Eck-LEDs einer Bedingung aus [Sch17]. Die Sortierung insgesamt, wird jedoch signifikant geändert, da die übergeordnete Sortierung nicht nach dem Haupt-Einfluss durchgeführt wird, sondern nach Abstand zu den Displayecken.

Zum Vergleich sind in Abbildung 6.21 die Ergebnisse der LED-Berechnung zwischen der Pixelorder-Sortierung nach Schäfer [Sch17] und der in diesem Abschnitt vorgestellten Sortierung zur Verbesserung des Schwarzwertes in den Displayecken gegenüber gestellt. Dazu wurde das HMI-Testbild 3 aus Anhang B genutzt.

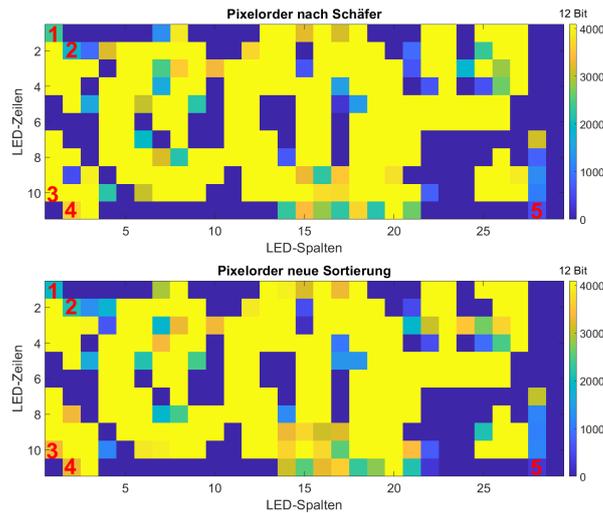


Abbildung 6.21.: Resultierende LED-Werte im Vergleich zwischen SoA nach Schäfer (oben) und der hier vorgestellten Methode zur Verbesserung des Schwarzwertes in den Displayecken (unten) für das HMI Testbild 3 (siehe Anhang B).

	markierte Positionen				
	1	2	3	4	5
SoA nach [Sch17]	57,4 %	44,3 %	100 %	90,1 %	13,1 %
neue Methode	46,8 %	51,5 %	81,8 %	79,9 %	6,9 %
Abweichung zu SoA	-18,4 %	16,2 %	-18,2 %	-11,3 %	-47,3 %

Tabelle 6.1.: LED-Werte (Duty-Cycle) an den markierten Positionen aus Abbildung 6.21.

In der Abbildung sind zudem 5 LEDs markiert, welche in Tabelle 6.1 verglichen werden. Es wird der jeweils berechnete LED-Wert (normiert auf das Maximum bzw. Duty-Cycle) an den markierten Positionen dargestellt. Die letzte Zeile zeigt die resultierende Abweichung in Prozent von der SoA Methode. Eine negative Abweichung ist gleichbedeutend mit einer dunkleren LED und somit ein besserer Schwarzwert in der Nachbarschaft. Es wird deutlich, dass die Positionen 1, 3, 4, 5 deutlich dunkler berechnet werden und den Schwarzwert verbessern. Die Position 2 jedoch wird heller als in SoA, da in diesem Fall die dunklere Eck-LED (Markierung 1) etwas kompensiert werden muss, um die gewünschte Leuchtdichte zu erzeugen. Insgesamt wird das Powersaving mit der neuen Sortierung nur um 0,1 Prozentpunkte schlechter. Somit ist die Sortierung der Pixelorder nach dieser neu vorgestellten Methode tragbar und kann den Schwarzwert alleine durch die neue Scanning-Sequenz der Pixelorder verbessern.

### 6.3.2. Parameterwahl

Unabhängig von der oben vorgestellte Methode zur Sortierung der Pixelorder können Local-Dimming Parameter angepasst werden, um den Fokus auf verbesserten Schwarzwert in den Displayecken zu verschieben. Bei der Anpassung der Pixelorder handelt es sich um feste, also *offline* berechnete Daten, welche zur Laufzeit in Hardware nicht geändert werden können. Die Parameter hingegen lassen sich auch zur Laufzeit einstellen und beeinflussen das Resultat des Optimizers direkt. Diese lassen sich also auch je nach Betriebsmodus zum

Beispiel Tag vs. Nacht je nach Priorität anpassen. Sie können auch spezifisch eingestellt werden.

### Begrenzung der Anzahl der inkrementierbaren LEDs (-Werte)

Bei der Berechnung der LED-Werte kann es vorkommen, dass eine einzige LED nicht ausreicht um die gewünschte Leuchtdichte (Condenserwert) in einem Pixel zu erzeugen. In diesem Fall werden – sobald die LED mit dem höchsten Einfluss ihr Maximum erreicht hat – weitere LEDs erhöht (inkrementiert). Im Hardware Design sind maximal 5 LEDs vorgesehen. Diese Anzahl kann durch Parameter reduziert werden. Ein Beispiel wird in Abschnitt 6.3.4 gezeigt.

### Anpassung des LED-Faktors an den Ecken

Eine weitere Möglichkeit den Schwarzwert in den Ecken zu verbessern, ist die Erhöhung der erlaubten Abweichung zwischen zwei benachbarten LEDs in horizontaler und vertikaler Richtung. Diese räumlichen Filter wurden in Abschnitt 6.2.2.1 eingeführt, um die Schwere des *Halo* Artefakts zu reduzieren. Hier muss also ein Trade-Off getroffen werden, zwischen Reduzierung des Halos und der Verbesserung der dunklen Ecken des Displays. Klar ist, dass der Parameter von der BLU als auch vom LC-Panel abhängig gewählt werden muss, da die Gefahr von Halos bei Erhöhung der erlaubten Abweichung zwischen zwei LEDs ansteigt. Alternativ kann die Änderung des LED-Faktors auch *lokal* nur für bestimmte LED-Positionen wie beispielsweise Displayecken appliziert werden, um die Halo-Unterdrückungsperformance im übrigen Bereich nicht zu beeinträchtigen.

### 6.3.3. Corner-Dimming

Um den Schwarzwert noch weiter zu verbessern, kann der Condenserinhalt ( $c$ ), also die tatsächlichen Condenserwerte an den Displayecken manipuliert werden. Das Ziel dieser Methode ist die künstliche Absenkung der Condenserwerte und die resultierende Beeinflussung der LED-Lösungen. Dazu wird eine Funktion  $r(d, APL)$  definiert, welche in Abhängigkeit des Abstandes  $d(c_{i,j})$  einer Condenserezelle  $c_{i,j}$  von den vier Ecken und des APLs einen Subtrahenden (günstig bzw. effizient in Hardware) oder einen Faktor zur Verrechnung mit dem Condenserwert  $c(i, j)$  liefert [SAX20]. Somit ergibt sich für den modifizierten Condenserwert  $c^r(i, j)$  und dem Operator OP, welcher als Subtraktion oder Multiplikation ausgeführt werden kann zu

$$c^r(i, j) = \begin{cases} \text{OP}(c(i, j), r(d, APL)), & d(c_{i,j}) \leq d_{\text{th}} \\ c(i, j), & d(c_{i,j}) > d_{\text{th}} \end{cases}, \quad \forall (i, j) \in \{0, \dots, N_C^r - 1\} \times \{0, \dots, N_C^c - 1\} \quad (6.2)$$

Um die Qualität von helleren, natürlichen Bildern nicht zu verschlechtern, kann in Abhängigkeit der APL in den Bereichen die Funktion bis auf  $r(d, APL) = 0$  bei einer Subtraktion bzw. auf 1 bei einer Multiplikation angepasst werden. Diese Funktion wird für die Hardwareimplementierung mit einem ROM beschrieben (als SRAM realisiert) und wird in die Berechnung der LED-Werte im Optimizer integriert. In Abbildung 6.22 ist ein Beispiel dieser Funktion für die Anwendung bei einer Subtraktion abgebildet. Es wird ein Ausschnitt von  $12 \times 12$  Condenserpositionen der linken, oberen Ecken gezeigt. Die z-Achse zeigt dabei den Subtrahenden mit einer 8 Bit Auflösung. Die Anwendung dieser Funktion auf den Condenserinhalt eines Testbilders ist in Abbildung 6.23 dargestellt. Es ist zu erkennen, dass einige der vormals hellen Zellen in den Ecken etwas abgedunkelt werden und wirkt direkt auf die Berechnung der LED-Werte, da der Condenserinhalt die Zielfunktion darstellt. Basierend auf diesen Condenserwerten ergeben die die LED-Werte wie in Abbildung 6.24

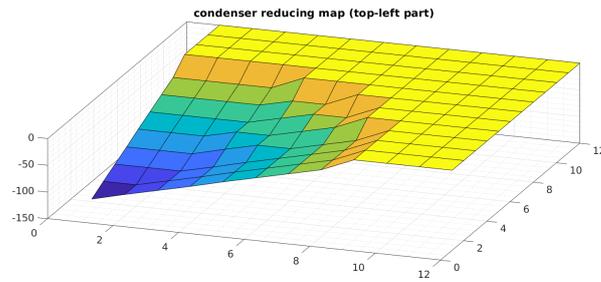


Abbildung 6.22.: Realisierungsmöglichkeit der Funktion  $r(d, APL)$  zur Anwendung auf die linke obere Ecke des Condensers.

abgebildet wird. Die Auswirkungen der Modifikation sind deutlich zu erkennen. Die LEDs an den Ecken werden somit dunkler und der Schwarzwert entsprechend besser. Die Gefahr von Clipping steigt zwar; da jedoch die vier Ecken im Allgemeinen die Grauwerte 0 oder 255 besitzen und keine Sicherheitsrelevanz besitzen, ist dies unproblematisch. Ein begrenztes Maß an Clipping ist jedoch akzeptabel [Alb10; Jun14; Sch17; KK09]. Die reduzierte Leuchtdichte an den vier Ecken bildet einen sanften Übergang zum dunklen Displayrahmen oder Cockpit-Design im Automobil und wirkt somit nahtlos integriert.

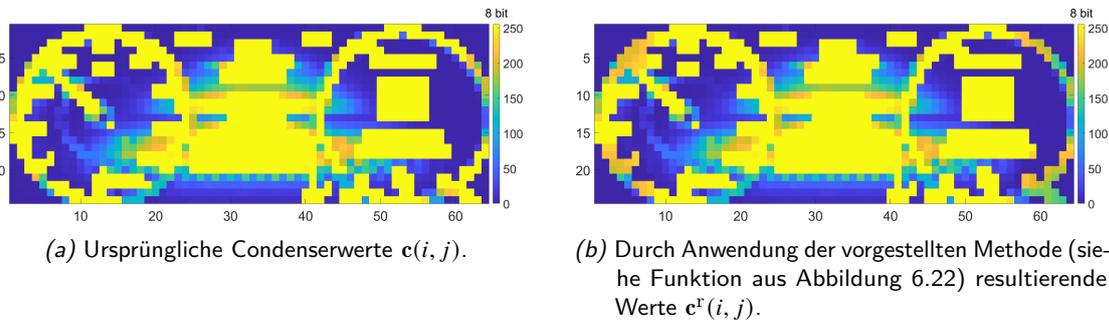


Abbildung 6.23.: Vergleich des Condenserinhalts mit und ohne Manipulation zu einem HMI-Testbild aus Anhang B.

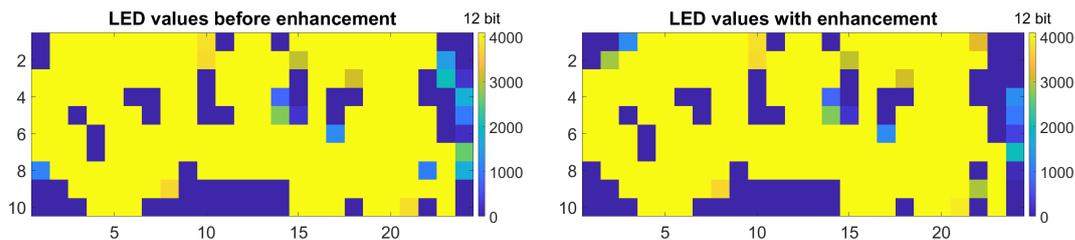


Abbildung 6.24.: Resultierende LED-Werte (einer 240 BLU) basierend auf Abbildung 6.23. Links ohne Modifizierung, rechts mit Adaption durch  $r(d, APL)$ .

### 6.3.4. Ergebnisse der Parameterwahl und Corner-Dimming

Außer der Umsortierung der Pixelorder, wurden hier weitere Methoden vorgestellt, um den Schwarzwert in den Displayecken weiter zu verbessern. Es handelt sich um die Parameterwahl sowie um das Corner-

Dimming. Es lässt sich feststellen, dass die Kombinationen der Methoden am effektivsten den Schwarzwert verbessern kann. Im Folgenden sei das Testbild 26 aus Anhang B betrachtet. Die Methode 1 entspricht die parametergesteuerte Reduktion der inkrementierbaren LEDs und Methode 2 sei die Anpassung des räumlichen Filters. Methode 3 ist das Corner-Dimming. Zum Vergleich werden die Methoden wie folgt kombiniert:

- Paramset default: Standardparameter ohne Änderung
- Paramset 1: Methode 2 und 3 kombiniert
- Paramset 2: Methode 1, 2 und 3 kombiniert

Die hier vorgestellten Methoden wurden in Hardware algorithmseitig implementiert und die Resultate in Form der Leuchtdichte anhand des oben genannten Testbildes mit der Eldim UMaster gemessen. Die Ergebnisse der Messung des Backlights sind in Abbildung 6.25 dargestellt. Es handelt sich um die Leuchtdichteinstellungen von  $400 \text{ cd m}^{-2}$ . Die Werte der Farbskala sind in  $\text{cd m}^{-2}$ . Die mittlere Spalte zeigt das reine gemessene Backlight mit den oben genannten Parameter. Die linke Spalte zeigt das Backlight mit einem Kontur-Plot, welche die Bereiche gleicher Leuchtdichte zur Verdeutlichung zur Geltung bringen. In der mittleren Spalte sind rote, gestrichelte Linien eingezeichnet. Diese fängt in einem optimalerweise schwarzen Bereich an.

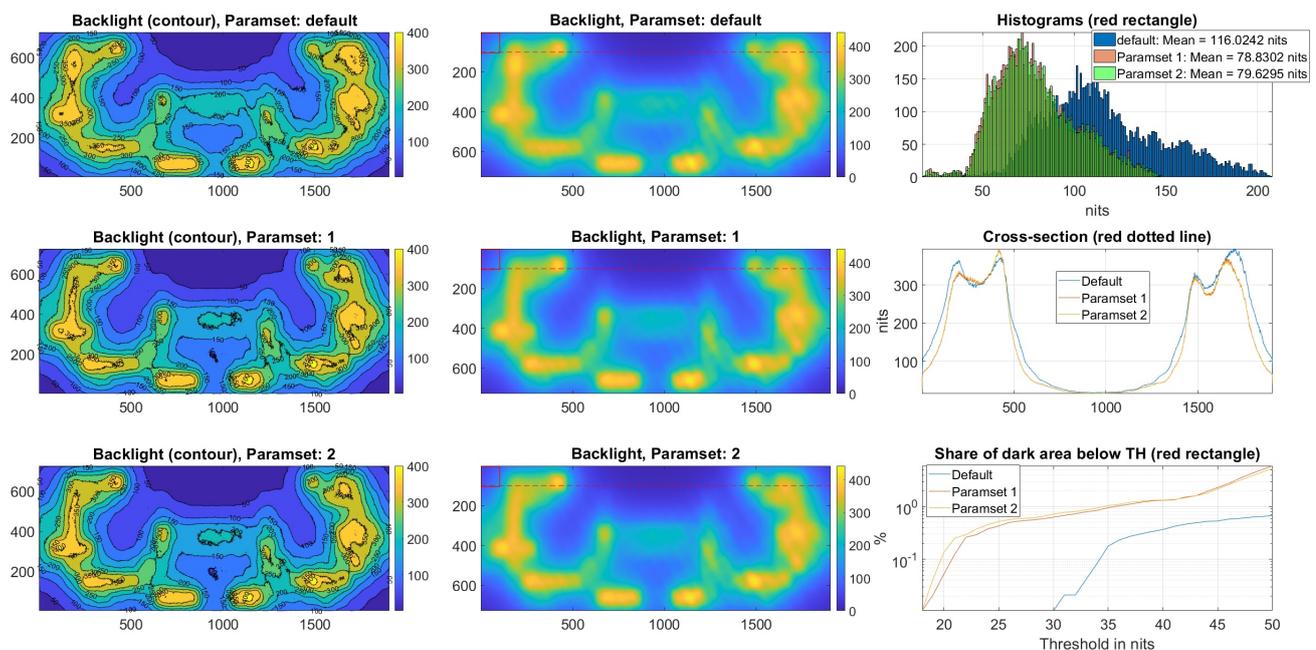


Abbildung 6.25.: Leuchtdichtemessungen der Schwarzwertverbesserung in den Displayecken durch Parameteranpassung und Corner-Dimming. Teilweise (Bildausschnitt) auch in [SAX20] veröffentlicht.

Entlang dieser Linien wird in der rechten Spalte, mittig ein Querschnitt durch alle drei Parameter-Sets aufgetragen und miteinander verglichen. Es lässt sich feststellen, dass durch die Anwendung der Methoden die resultierende Leuchtdichte zu den Rändern hin geringer wird. Zusätzlich sind noch die Ecken jeweils mit einem roten Quadrat gekennzeichnet. Der Inhalt, also die gemessene Leuchtdichte innerhalb der Quadrate

wird in Form eines Histogramms überlagert. Es ist zu erkennen, dass die gesamte Verteilung weiter in den dunklen Bereich verschoben wird. Der prozentuale Anteil der schwarzen Fläche innerhalb der Quadrate wird unten rechts visualisiert. Es ist der Anteil aufgetragen, welcher dunkler als auf der x-Achse aufgetragene Grenzwert ist. Im Standardfall (blau) sind beispielsweise keine Anteile unter  $30 \text{ cd m}^{-2}$  vorhanden. Es wird somit ersichtlich, welchen Einfluss die Parameteränderungen auf den Schwarzwert haben. Die Anteile dunkler Flächen wird größer und es wird somit eine nahtlose Integration ermöglicht. Die visuelle Qualität kann folglich mit diesen Methoden weiter gesteigert werden. Diese Methoden führen ein weiteres sichtbares Differenzierungsmerkmal ein und erhöhen somit speziell die Wertigkeit des Gesamtsystems in Kombination mit dem Interior Design.

## 6.4. Erhaltung und Sicherstellung der visuellen Qualität bei LED-Ausfall

Um eine gute Qualität und sicheren Betrieb zu ermöglichen, muss auch der mögliche Ausfall eines LED-Stranges betrachtet werden und weitere Qualitätsmetriken eingeführt werden. Zudem existiert eine International Organization for Standardization (ISO)-Norm *ISO 26262* [ISO18] für die *funktionale Sicherheit* in Kraftfahrzeugen. Diese wird bei sicherheitsrelevanten System angewandt. Im Falle eines ICDs ist die Erfüllung erwünscht. Je nach Risiko, welches die Eintrittswahrscheinlichkeit eines bestimmten Fehlers und deren Wirkung auf die Umwelt, bzw. den möglichen Schaden enthält, wird ein System in verschiedene Automotive Safety Integrity Level (ASIL) Klassen eingestuft. So kann beispielsweise der Ausfall eines LED-Stranges dazu führen, dass ein Mindestkontrast [DIN17b] unterschritten wird und Warnsymbole zu spät erkannt werden. Ein weiteres Beispiel ist das Einfrieren des Bildes und der mögliche Schaden als Resultat nicht korrekten Instrumentenanzeigen. Der hier vorgestellte Local-Dimming hat das USP, der Abwesenheit von Frame-Buffern und kommt somit einer Zertifizierung der IP zu Gute. Kommt es zu einem externen Ausfall des Pixelstreams durch das IP, so besteht keine Gefahr eines Freezes, da kein Framebuffer vorhanden ist. Der Fahrer erkennt durch die Abwesenheit des Bildes (schwarz, keine Information), dass ein Fehler vorliegt. Folglich kann es nicht zu fatalen Fehlannahmen durch eine falsche Anzeige führen. Im nächsten Abschnitt wird auf den LED-Ausfall eingegangen und eine Methode zur Entschärfung vorgestellt.

Als qualitäts-erhaltende Maßnahme wird der Ausfall einer LED vorgestellt. In Abbildung 6.26 sind dazu Messungen der 133er BLU dargestellt, welche das Problem besonders gut zeigt. Linkes Bild weist keine Defekte auf, während rechts zwei LED-Defekte vorhanden sind. Die Ausfälle sind deutlich zu erkennen. Insbesondere jedoch der Ausfall in der fünften LED-Zeile. Im Rahmen dieser Arbeit wurde ein weiteres

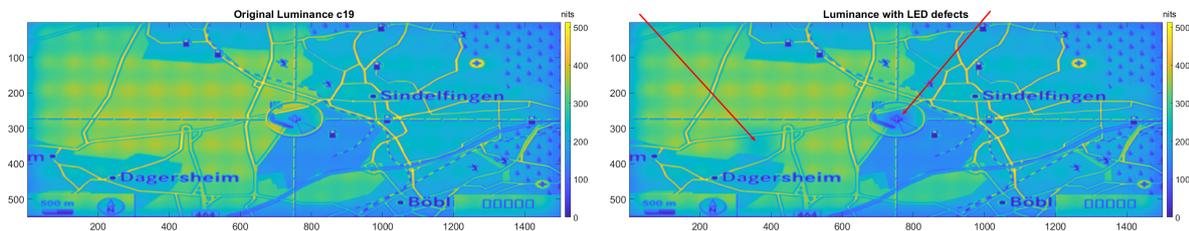


Abbildung 6.26.: Leuchtdichteaufnahmen einer 133er BLU im Ursprungszustand (links) und mit Ausfall zweier LEDs (rechts). Der Ausfall ist einerseits genau in der Bildmitte und andererseits in der LED-Zeile und Spalte 5.

Constraint eingeführt, welches die Berücksichtigung von ausgefallenen LEDs während der Optimierung erlaubt. Somit wird auch die Pixelkompensation korrekt durch das berechnete Backlight durchgeführt. Der Defekt einer LED kann durch den LED-Treiber detektiert, bzw. an das IP kommuniziert werden. Die meisten modernen Treiber haben dazu eine Short- bzw. Open-Erkennung um einen LED-Defekt zu detektieren [Tex14]. Diese Information kann dann seriell, beispielsweise per SPI übertragen werden. In der IP angekommen, werden die defekten LEDs mit einem Bit markiert und die entsprechenden LEDs so eingeschränkt, dass ein Maximalwert  $v_{\max}$  nicht überschritten werden kann. Dieser kann auch in Abhängigkeit von der APL und weitere Bildmerkmale gewählt werden. Zudem wird die Anzahl inkrementierbaren LEDs um die defekte LED herum erhöht (siehe Abschnitt 6.3.2) um eine höhere Leuchtdichte bereitstellen zu können. Wird eine defekte LED als Haupt-LED während der Bearbeitung der Pixelorder betrachtet, so wird die nachrangige LED erhöht, sobald die defekte LED den Wert  $v_{\max}$  erreicht hat. Insbesondere für  $v_{\max} > 0$ , kann das Clipping, welches durch die fehlende Leuchtdichte verursacht wird auch mit der *Clipping Suppression Function* aus [SJX12; Sch17] unterdrückt werden. Hierbei wird ein Abrutschen ins Clipping verhindert, jedoch die Leuchtdichte etwas reduziert.

In Abbildung 6.27 wird wieder eine Leuchtdichtemessung zum selbigen Backlight und Testbild aus Abbildung 6.26 im Ausschnitt dargestellt. Hierbei wurde links keine Kompensation des Ausfalls durchgeführt, während er rechts durch die Constraints im Local-Dimming Prozessor bei der LED-Wert Berechnung berücksichtigt wird. Zum Vergleich sind dazu die Querschnitte durch das ursprüngliche Bild und durch das Resultat (jeweils mit gestrichelten vertikalen Linien markiert) im unteren Teil dargestellt. Es ist erkennbar, dass die Berücksichtigung der defekten LEDs dazu führt, dass die Leuchtdichte um die defekte LED herum deutlich höher liegt als im nicht kompensierten Fall. Dabei wird  $> 85\%$  der Zielleuchtdichte erreicht [SAX20]. Somit wird deutlich, dass die durch den LED-Ausfall verursachte Leuchtdichtelücke mit dieser Methode deutlich reduziert werden kann. Die IP kann mehrere LED-Ausfälle berücksichtigen. Im Ernstfall reduziert

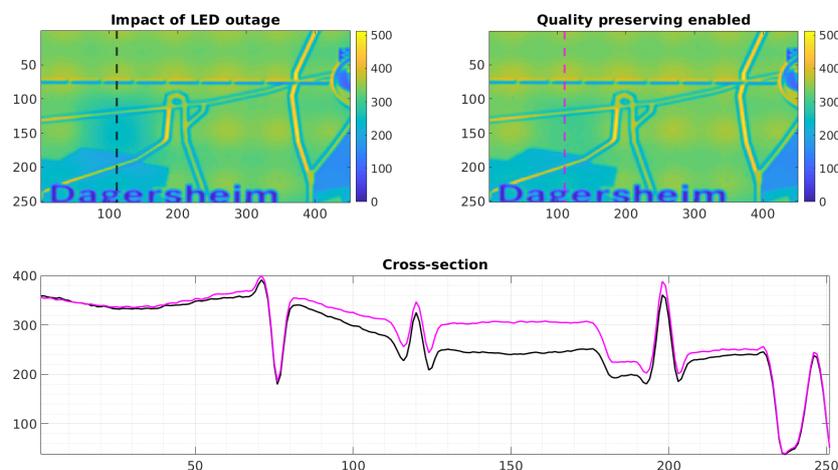


Abbildung 6.27.: LED-Ausfall einer LED mit und ohne Gegenmaßnahme. Links Ausfall ohne Kompensation, rechts mit Gegenmaßnahme. Entnommen aus [SAX20].

sich Powersaving leicht, wobei die Qualitätseinbußen durch den LED-Ausfall gut kompensiert werden können. Somit ist diese Methode gut geeignet um eine Einordnung in eine höhere ASIL Klasse zu begünstigen. Insbesondere im Falle mehrerer Defekte werden die Stärken des Systems noch weiter ausgenutzt. Da Displays im Automobil eine immer wichtigere Rolle einnehmen, ist es gut vorstellbar, dass noch höher-auflösende BLUs in Zukunft verwendet werden, um die Bildqualität weiter zu steigern. Bereits in

Abschnitt 5.1.2 wurde die Limitierung der verfügbaren Berechnungszeit deutlich. Insbesondere bei weiter steigenden LED-Auflösungen müssen weitere Methoden eingeführt werden, um diese Constraints zu erfüllen. Im nächsten Abschnitt wird darauf eingegangen.

## 6.5. Methoden zur Abdeckungen noch höherer BLU-Auflösungen

Wie beschrieben ist die gesamte Berechnungszeit  $t_{\text{total}} = t_{\text{LEDs}} + t_{\text{SP}}$  eine Funktion der Form  $t_{\text{total}} = f(N_L, \tilde{N}_L, N_{\text{iter}}, N_C, N_{\text{SP}}, f_s)$ . In Abbildung 6.28 ist die genannte Berechnungszeit in Abhängigkeit von der Quadratwurzel der LED-Anzahl dargestellt. Angenommen werden hier 1536 Condenserezellen ( $N_C$ ), 6321 Stützstellen ( $N_{\text{SP}}$ ) sowie eine Systemclock mit einer Frequenz von  $f_s = 200$  MHz. Wie in Abschnitt 5.2.2 erläutert wurde, wird durch die neue Methode, nur ein Teil der lichtbeitragenden LEDs bei der Berechnung berücksichtigt. Dadurch bleibt  $t_{\text{LEDs}} = \text{const.}$  unabhängig von der LED-Anzahl (in Blau dargestellt), während  $t_{\text{SPs}}$  linear mit  $N_L$  ansteigt. Mit gestrichelten Linien sind jeweils die Framerate für eine Bildwiederholfrequenz von 60 Hz bzw. 75 Hz markiert und die resultierenden Obergrenzen der LED-Anzahl können abgelesen werden. Zum Vergleich wurden in Abbildung 5.5 auf Seite 133 bereits die Berechnungszeiten für den SoA SSC-Algorithmus nach [Alb10; Jun14; Sch17] abgebildet. Steigt die Anzahl der LEDs ( $N_L$ ) oder die

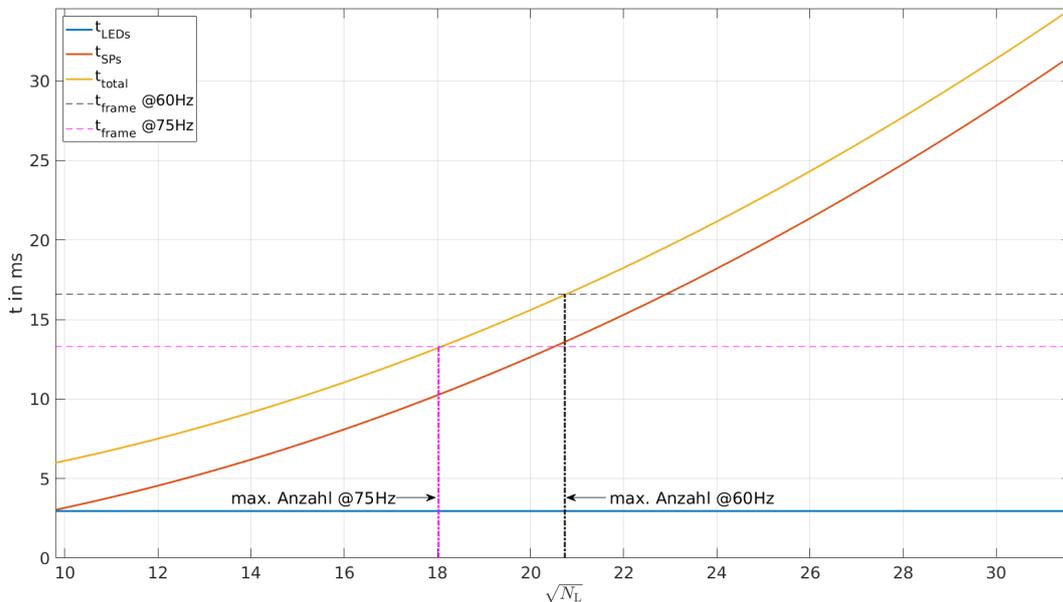


Abbildung 6.28.: Berechnungszeiten in Abhängigkeit der Quadratwurzel der LED-Anzahl für die aktuellen Parameter des auf den Prototypen implementierten Algorithmus.

Anzahl der benötigten Condenserezellen ( $N_C$ ), so überschreitet die benötigte Berechnungszeit schnell die Dauer eines Frames ( $t_{\text{LEDs}} + t_{\text{SP}} > t_{\text{frame}}$ ). Für diese Fälle müssen weitere Schritte zur Reduktion der Gesamtberechnungszeit unternommen werden, um den Algorithmus realisierbar zu halten.

In Abbildung 6.29 wird ein Diagramm gezeigt, welches die Abhängigkeiten der verschiedenen Parameter auf die Berechnungszeit darstellt. Die oberen roten Parameter, also die Pixel- und LED-Anzahl seien vorgegeben und nicht änderbar. Diese beeinflussen zusätzlich die übrigen, in Grün dargestellten Parameter in der oberen Reihe (nicht durch Pfeile angedeutet), welche an die Gegebenheiten angepasst werden können. Insgesamt entsteht so ein mehrstufiger Abhängigkeitsbaum. In den nächsten Abschnitten werden einige Methoden

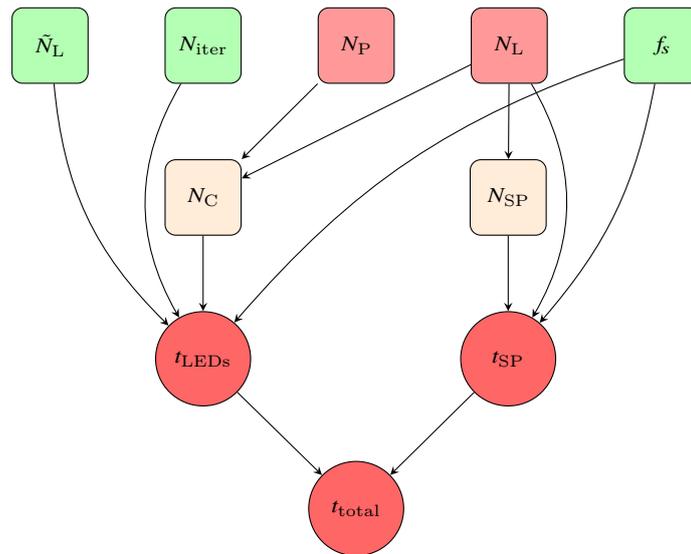


Abbildung 6.29.: Abhängigkeitsdiagramm der Berechnungszeit  $t_{\text{total}}$ .

vorgestellt, um die beiden Zeiten unabhängig voneinander zu reduzieren, sodass höhere BLU-Auflösungen prozessiert werden können.

### 6.5.1. Erhöhung der Systemfrequenz

Da die Module bzw. ihre Registerstufen und Logik für die Berechnungen mit der Systemclock der Frequenz  $f_s$  getaktet werden, ist ein möglicher Ansatz diese Frequenz weiter zu erhöhen. Für die beiden Berechnungszeiten gilt:

$$t_{\text{LEDs}} \propto \frac{1}{f_s} \text{ sowie } t_{\text{SP}} \propto \frac{1}{f_s}$$

Dabei ist die maximale Frequenz von der zugrundeliegenden Strukturgröße abhängig. Auch spielt es eine Rolle, ob es sich um einen ASIC oder FPGA handelt. Generell gilt, je kleiner die Struktur, desto höher kann getaktet werden, da die einzelnen Gatter eine kürzere Distanz und somit Signallaufzeit untereinander haben. Diese Methode kann auch in Kombination mit weiteren Optimierungen genutzt werden. In Abbildung 6.30 ist dieses Verhalten mit logarithmischer Ordinate dargestellt. Gezeigt werden die frequenzabhängigen Berechnungszeiten  $t_{\text{LEDs}}$ , welche *unabhängig* von der konkreten LED-Anzahl  $N_L$  ist. Daneben sind noch die Berechnungszeiten  $t_{\text{SPs}}$  für eine LED-Anzahl von 240 bzw. 1080 dargestellt. Aufgrund der reziproken Proportionalität sind die Berechnungszeiten für kleinere Frequenzen sehr hoch und überschreiten die Framedauer. Ebenfalls erkennbar ist die 4,5-Fachung von  $t_{\text{SPs}}$  bei  $N_L = 1080$  im Vergleich zu  $N_L = 1080/4,5 = 240$  LEDs.

Diese Methode beeinflusst die beiden Berechnungszeiten einzig durch Erhöhung der Frequenz und ist in einem gewissen – von der Hardware abhängigen Rahmen – gut umzusetzen. So würde eine Verdoppelung der Frequenz die Halbierung von  $t_{\text{SPs}}$  herbeiführen und somit die doppelte Anzahl von LEDs erlauben. Diese Methode stößt jedoch schnell an ihre Grenzen, da eine Erhöhung der Frequenz ab einem gewissen Punkt nur noch schwer realisierbar ist. Folglich werden noch Methoden vorgestellt, welche die Berechnungszeiten bei  $f_s = \text{const.}$  verringern können. Dabei kann jeweils  $t_{\text{LEDs}}$  sowie  $t_{\text{SPs}}$  unabhängig voneinander reduziert werden. Die Berechnungszeit  $t_{\text{LEDs}}$  kann durch die Parallelisierung der Pixelorder wie in [Alb10; Jos20] gezeigt durchgeführt werden. Eine deutlich höhere Reduktion die Berechnungszeit kann jedoch bei  $t_{\text{SPs}}$

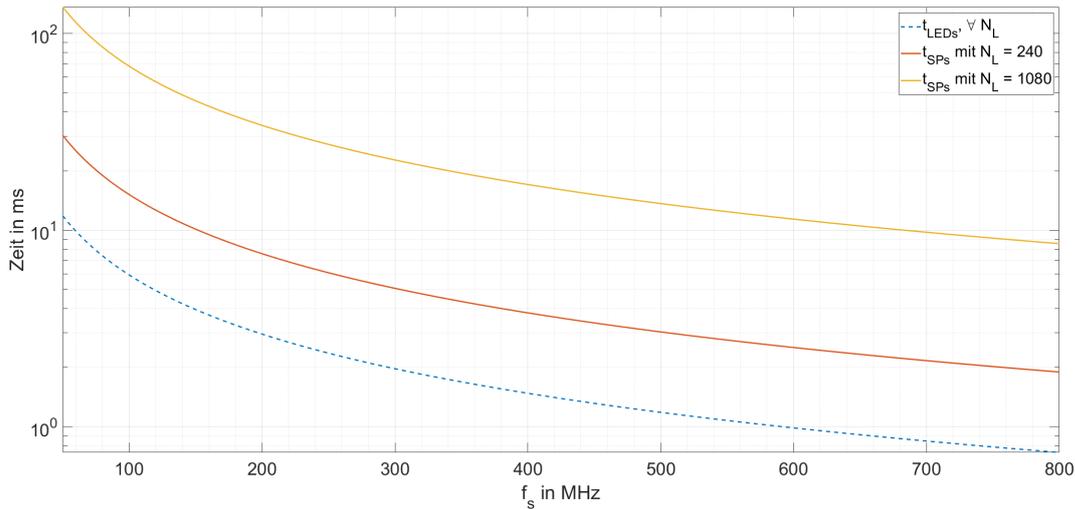


Abbildung 6.30.: Logarithmische Darstellung der Berechnungszeiten  $t_{LEDs}$  sowie  $t_{SPs}$  für  $N_L = 240$  und 1080.

erreicht werden, wie im nächsten Abschnitt vorgestellt wird.

### 6.5.2. Betrachtung einer begrenzten LED-Zahl zur Sample-Point-Berechnung

Aus den vorangegangenen Abschnitten ging hervor, dass die Berechnungszeit  $t_{SP}$  für eine höhere LED-Anzahl ( $N_L$ ) optimiert werden soll, sodass  $t_{total}$  weiter reduziert werden kann. Zu diesem Zweck werden im Folgenden zwei Methoden vorgestellt, welche zu einer Reduktion von  $t_{SP}$  führen.

Da eine Erhöhung der LED-Anzahl die SP-Berechnungszeit  $t_{SPs}$  verlängert, zielt diese Methode darauf ab, weniger LEDs bei der Berechnung der Sample-Points zu berücksichtigen. Dadurch gilt  $t_{SPs} = const.$  und wird damit unabhängig von den tatsächlich vorhandenen  $N_L$  LEDs. In Abschnitt 5.2.2 wurde bereits diese Methode für die Berechnung der LED-Werte vorgestellt und verhält sich hier analog. In Abbildung 5.7 auf Seite 137 ist das Konzept gezeigt und lässt sich so auch auf die Berechnung der Sample-Points übernehmen.

$$\check{\mathbf{B}}^{SP} := \left( \check{b}_{r,c}^{SP} \right) = \sum_{l \in \mathcal{M}_{r,c}} \check{L}_{(r,c),l}^{SP'} \cdot \mathbf{v}(l) \quad (6.3)$$

Dabei ist  $\mathcal{M}_{r,c}$  hier die Menge der LEDs (bzw. der LED-Nummern) um den Sample-Point  $(r, c)$  welche die direkte Nachbarschaft um die zugehörige Haupt-LED repräsentieren. Für die Evaluierung werden die 133er BLU sowie eine 240er BLU der vorhandenen Prototypen verwendet. Dazu werden verschiedene Beschränkungen der berücksichtigten LEDs verglichen. Den Mittelpunkt stellt immer die zu einer Condenserzelle gehörige Haupt-LED dar. Somit lässt sich eine LED-Umgebung der direkten Nachbarschaft von  $(2n + 1) \times (2n + 1)$  mit  $n \in \mathbb{N}$  LEDs nutzen. In diesem Abschnitt wird  $n \in \{1, 2, 3, 5, 7\}$  ausgewählt. Für die 133er BLU entspricht die Berücksichtigung von  $15 \times 15$  LEDs der Berücksichtigung aller vorhandenen LEDs. In Abbildung 6.31 wird das resultierende virtuelle Backlight aus Formel (6.3) für die beiden BLUs dargestellt. Die LED-Werte sind hier auf eine PWM von 100% eingestellt. Werden alle LEDs in jedem Sample-Point berücksichtigt, so gilt  $\check{\mathbf{B}}^{SP} = \mathbb{1}^{N_{SP}^r \times N_{SP}^c}$ . In der Abbildung lässt sich feststellen, dass die 240er BLU im Vergleich zur 133er BLU eine geringere Abweichung zur tatsächlichen normierten Leuchtdichte ( $\mathbb{1}$ ) liefert. Dies resultiert aus der Lokalität bzw. des Einflusses, also der eigentlichen LSF. Je globaler eine LSF ist, desto mehr LEDs müssen

berücksichtigt werden, um den Fehler gering zu halten. Der RMSE zum Optimum ist dazu in Abbildung 6.32 dargestellt und bestätigt diese Beobachtung. In Abbildung 6.31 lässt sich zudem feststellen, dass der Fehler an den Ecken bei weniger berücksichtigten LEDs kleiner ist als in der Mitte des Displays. Die Summe kommt also an den Ecken näher an die 1 heran. Mit zunehmender LED-Anzahl ändert sich jedoch dieses Verhalten und die SPs liefern innen-liegend die besseren Ergebnisse. Dieses Phänomen kommt dadurch zustande, dass die Einflüsse der LEDs in den Ecken immer größer sind als auf die innen-liegenden Pixel. Es reichen also weniger LEDs aus, um sich an den Ecken der 1 zu nähern (siehe auch Abschnitt 4.1). Werden also nur wenige LEDs betrachtet, erreichen die SPs an den Ecken bessere Werte. Kommen weitere LEDs dazu greift dieser Vorteil nicht mehr und die SPs aus dem Inneren erreichen nun genauere Ergebnisse. Zudem überlagert noch der Effekt, dass bei beispielsweise von den  $5 \times 5$  betrachteten LEDs bei den äußeren SPs nur insgesamt 9 LEDs genutzt werden. Die übrigen würden außerhalb des Displays liegen. Diese beschriebene Effekte sind bei der 133er BLU sehr gut beim Übergang von  $5 \times 5$  auf  $7 \times 7$  in der Abbildung 6.31 sichtbar. Zur weiteren Betrachtung werden die beiden im vorangegangenen Abschnitt

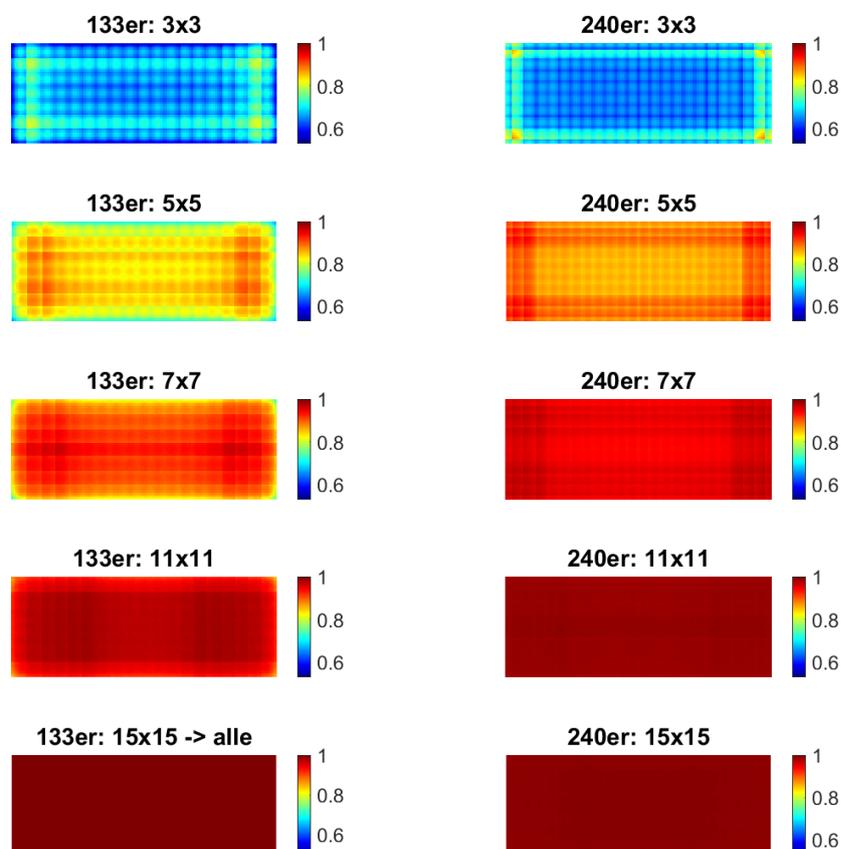


Abbildung 6.31.: Visualisierung des resultierenden  $\tilde{B}^{SP}$  bei Betrachtung einer beschränkten LED-Menge für eine 133er und eine 240er BLU.

verwendeten HMI-Testvideos genutzt und der RMSE zu den zugehörigen Frames berechnet. Das Resultat des *Testvideo 2* ist in Abbildung 6.33 für verschiedene Anzahl der berücksichtigten LEDs dargestellt (in Blau). Der APL ist in Rot aufgetragen. Der obere Teil der Abbildung zeigt dabei das Ergebnis der 133er BLU, der untere der 240er BLU. Der Fehler ist im Allgemeinen bei einer höheren APL im Durchschnitt höher und ist eine Folge der LSF und der Zielgröße (Condenserwert). Im Vergleich zur 133er BLU liefert die

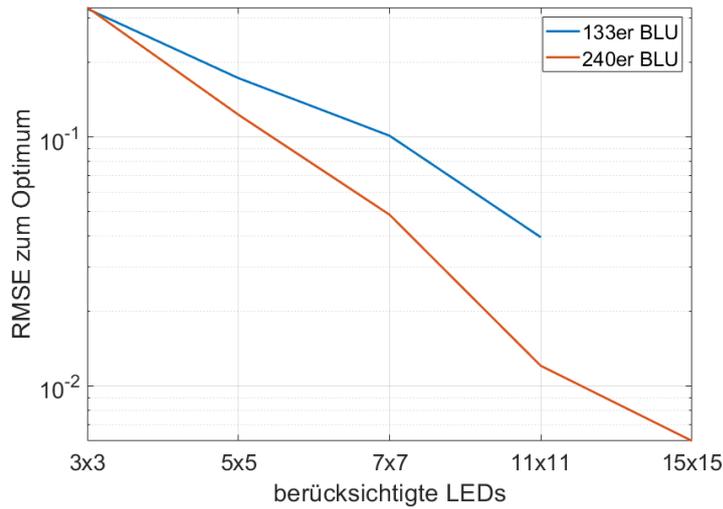


Abbildung 6.32.: Fehler des virtuellen Backlights bei Reduktion der betrachteten LEDs pro Sample-Point zu Abbildung 6.31. Bei der 133er BLU beträgt der Fehler bei  $15 \times 15$  LEDs 0 und somit nicht in der logarithmischen Skala dargestellt.

240er einen deutlich geringeren RMSE. Beispielsweise liegt der Fehler bei der 240er bei einer Betrachtung von  $5 \times 5$  LEDs einem ähnlichen Niveau wie die Betrachtung von  $7 \times 7$  LEDs bei der 133er BLU. Folglich kann mit der 240er BLU die Berechnungszeit bei selbem Fehler weiter reduziert werden.

### Reduktion des Fehlers durch Korrektur

Die vorgestellte Methode der Reduktion der betrachteten LEDs bei der Sample-Point Berechnung führt zu einer Abweichung von den tatsächlichen Sample-Points bei Betrachtung aller LEDs (vgl. Abbildung 6.33). Dieser resultierende Fehler kann jedoch mit verschiedenen Methoden reduziert werden. Die korrekten SP-Werte berechnen sich zu

$$\check{\mathbf{B}}^{\text{SP}} := \left( \check{b}_{r,c}^{\text{SP}} \right) = \sum_{l \in \mathcal{M}_{r,c}} \check{L}_{(r,c),l}^{\text{SP}'} \cdot \mathbf{v}(l) + \underbrace{\sum_{l \in \mathcal{L} \setminus \mathcal{M}_{r,c}} \check{L}_{(r,c),l}^{\text{SP}'}}_{\text{nicht berücksichtigt}} \cdot \mathbf{v}(l)$$

dabei ist  $\mathcal{L}$  die Menge aller LEDs. Der rechte Summand wurde also zum Zwecke der Berechnungszeit-optimierung nicht betrachtet und induziert die weiter oben berechneten Fehler. Wird nun das Optimum  $\check{\mathbf{B}}^{\text{SP}} = \mathbf{1}$  angenommen, so ergibt sich für die maximalen LED-Werte ein maximaler *Offset*, welcher aufgrund der Reduktion der Berechnungszeit nicht berücksichtigt wird. Dies wird in Formel (6.4) dargestellt.

$$1 - \sum_{l \in \mathcal{M}_{r,c}} \check{L}_{(r,c),l}^{\text{SP}'} = \sum_{l \in \mathcal{L} \setminus \mathcal{M}_{r,c}} \check{L}_{(r,c),l}^{\text{SP}'} = \underbrace{O_{r,c}^{\text{SP}}}_{\hat{=} \text{maximaler Offset}} \quad (6.4)$$

Um den Fehler trotz Limitierung der betrachteten LEDs zu reduzieren, kann der fehlende Summand als Funktion des maximalen Offsets geschätzt werden. Der Offset selbst, wie in Formel (6.4) gezeigt, lässt sich für jeden Sample-Point einmalig berechnen und bei Bedarf speichern. Somit ergibt sich ein Sample-Point wie in Formel (6.5) dargestellt.

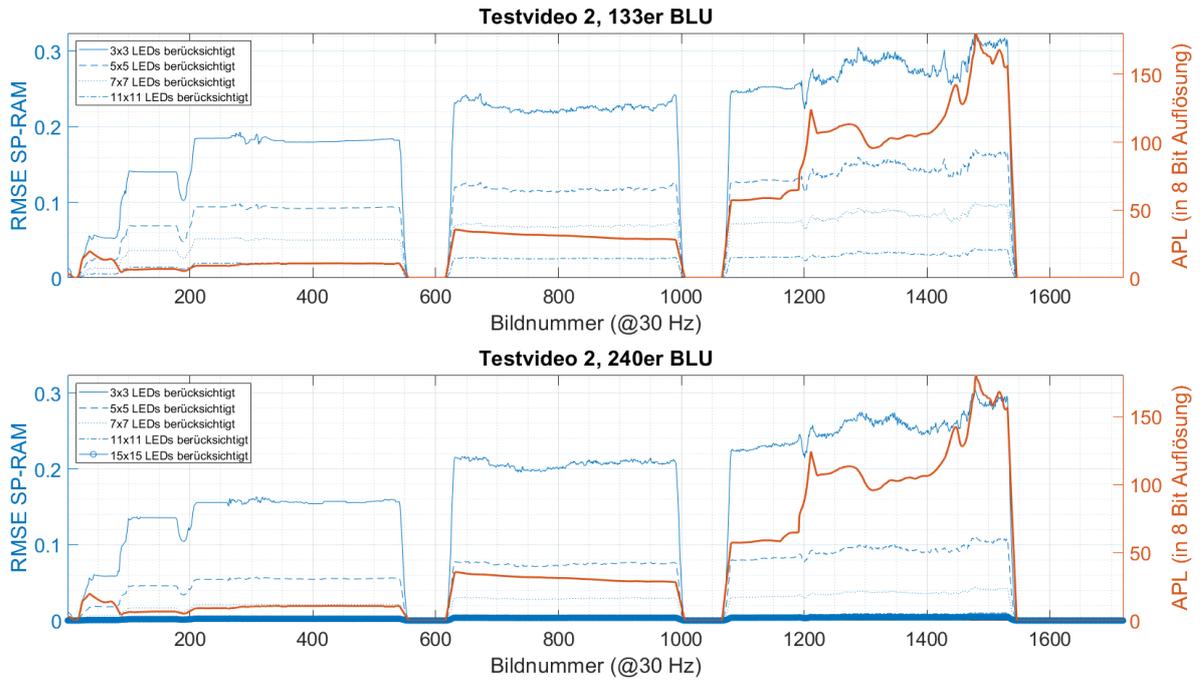


Abbildung 6.33.: Resultierender RMSE des berechneten Backlights durch Betrachtung einer definierten LED-Zahl für das Testvideo 2.

$$\check{b}_{r,c}^{\text{SP}} = \sum_{l \in \mathcal{M}_{r,c}} \check{L}_{(r,c),l}^{\text{SP}'} + \underbrace{f(O_{r,c}^{\text{SP}})}_{\text{Schätzung}} \quad (6.5)$$

Die entsprechende Funktion  $f$  kann je nach Hardwareaufwand ausgeführt werden. Beispielhaft werden hier zwei Methoden durch die Funktionen  $f_1$  und  $f_2$  verwendet. Diese sind wie in Formel (6.6) dargestellt realisiert.

$$\begin{aligned} f_1(O_{r,c}^{\text{SP}}) &= O_{r,c}^{\text{SP}} \\ f_2(O_{r,c}^{\text{SP}}) &= O_{r,c}^{\text{SP}} \cdot \text{APL}^\beta \end{aligned} \quad (6.6)$$

Dabei bezieht sich das APL auf das aktuelle Frame. Der Exponent  $\beta$  wird hier zu  $\beta = 3,5^{-1}$  gesetzt und gewichtet dabei den maximalen Offset mit dem aktuellen APL des Bildes. Die Methode 1 hingegen ist eine triviale Identitätsfunktion, welche dem Summanden gerade den einmalig berechneten maximalen Offset zuordnet.

Werden diese Methoden auf die Testvideos angewandt, so ergeben sich die Fehler für das *Testvideo 2* für die beiden BLUs wie in Abbildung 6.34 abgebildet. Es werden hier  $7 \times 7$  LEDs betrachtet. In Blau ist dabei der Fehler ohne Korrektur dargestellt. Rot ist die Methode 1 ( $f_1$ ) und in Orange die Methode 2 ( $f_2$ ). Auffallend ist der höhere Fehler mit  $f_1$  als die Referenz bei sehr dunklen Bildern. Bei schwarzen Bildern ist der Fehler jedoch nicht nachteilig, da die Pixelwerte 0 sind und somit die Kompensation keine Artefakte produziert. Das Problem bei dieser Methode ist jedoch, dass auch immer und unabhängig vom Bildinhalt gilt  $\check{b}_{r,c}^{\text{SP}} \geq O_{r,c}^{\text{SP}}$

und somit bei nicht schwarzen, aber dunklen Bildern der Sample-Point-Wert deutlich höher werden kann als der tatsächliche Wert bei Betrachtung aller LEDs und somit der Fehler größer wird. Die Methode 2 hingegen liefert einen deutlich geringeren Fehler und ist nur in einem Bruchteil der Frames größer als der Fehler ohne die Korrektur. Insgesamt gilt für die Betrachtung, dass zu *helle* Sample-Points dazu führen, dass die Pixel zu wenig kompensiert werden. Dadurch wird das Bild an diesen Stellen dunkler. Zu dunkle

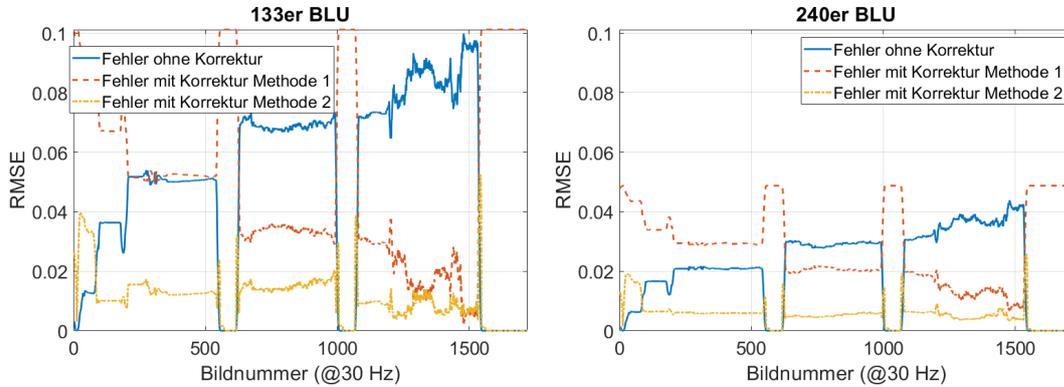


Abbildung 6.34.: Resultierender RMSE des berechneten Backlights durch Betrachtung von  $7 \times 7$  LEDs mit Offset-Korrektur pro Sample-Point bei Testvideo 2.

Sample-Points können jedoch zu Clipping führen. Dies gilt es zu vermeiden. Wie bereits oben gesehen, ist der Fehler bei der 240er BLU deutlich geringer als bei der 133. Die Fehler pro Frame für das *Testvideo 1* sind auf Abbildung A.9 auf Seite 246 abgebildet. Die über die beiden Testvideos gemittelten RMSE für die beiden BLUs und den beiden Korrekturmethode sind in Tabelle 6.2 dargestellt.

Es ist zu erkennen, dass der RMSE der Methode 1 sogar im Mittel stärker von der Referenz abweicht, als die unkorrigierte Variante. Es ist jedoch anzumerken, dass es in den Testvideos auch viele schwarze Frames gibt, welche einen hohen RMSE bei Methode 1 induzieren, jedoch im Sinne der Bildqualität nicht relevant sind. Die Methode 2 hingegen liefert für die 133er BLU einen um  $\approx 73\%$  geringeren Fehler als ohne Korrektur. Für das 240er ist der Fehler etwa 71% geringer. Im Vergleich zum 133er liefert die Methode 2 ein um  $\approx 55\%$  kleineren Fehler.

	keine	Korrekturmethode	
		Methode 1	Methode 2
133er BLU	4,66 %	5,55 %	1,28 %
240er BLU	1,99 %	2,98 %	0,58 %

Tabelle 6.2.: RMSE bei  $7 \times 7$  LEDs Betrachtung mit Fehlerkorrekturen der SP-Berechnung basierend auf den beiden Testvideos.

Ein Beispiel ist in Abbildung 6.35 abgebildet. Es werden die berechneten Sample-Points des Frames 450 aus Testvideo 2 gezeigt, basierend auf der 240er BLU. Im unteren Teil sind die Querschnitte zu durch die markierten Zeilen dargestellt. Gut erkennbar ist, dass Methode 1 hier größere Werte liefert als die Referenz. Wird keine Korrektur angewandt, so sind die Werte tendenziell zu gering und führen zu einer Überkompensation womit das Clipping-Risiko gesteigert wird. Das Resultat der Methode 2 hingegen unterscheidet sich nur marginal von der Referenz.

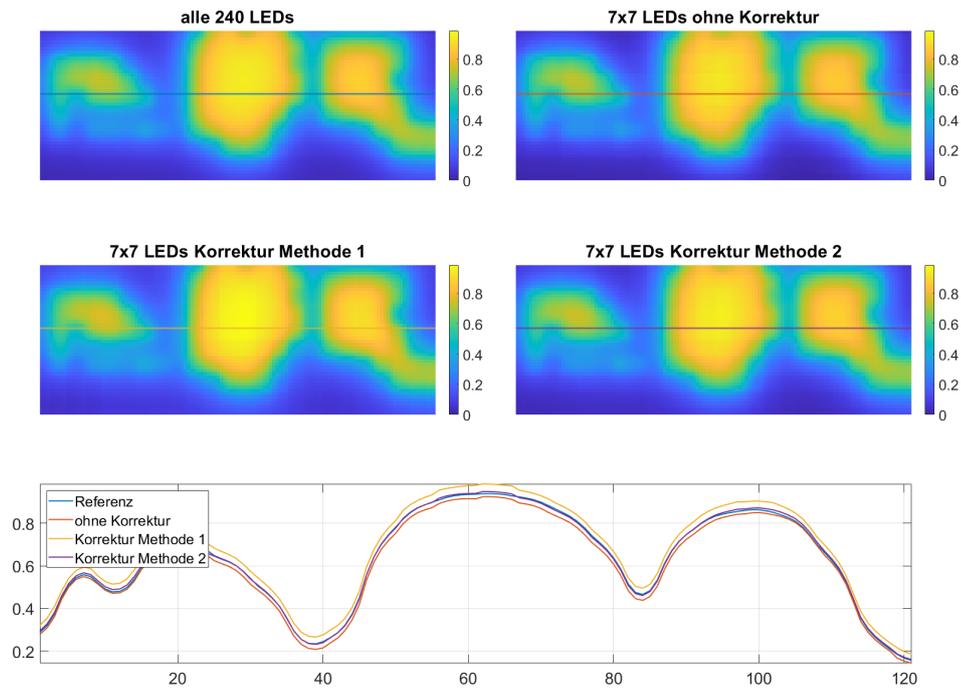


Abbildung 6.35.: Resultierende  $\tilde{B}^{\text{SP}}$  des Frames 450 von Testvideos 2 mit und ohne Korrekturen der Sample-Points.

### 6.5.3. Resultate

In diesem Kapitel wurden Methoden vorgestellt, welche dazu dienen die Berechnungszeiten  $t_{\text{LED}} + t_{\text{SPs}} = t_{\text{total}}$  zu reduzieren, um unterhalb der Framedauer  $t_{\text{frame}}$  zu bleiben. Einen Vergleich für ein Beispiel mit  $N_{\text{L}} = 1080$  ist in Abbildung 6.36 dargestellt. Dazu seien drei Fälle im Beispiel unterschieden

- Fall 1: neue Implementierung ohne weitere Reduktion der Berechnungszeiten
- Fall 2: SP-Berechnung mit verringerter LED-Anzahl (Abschnitt 6.5.2)
- Fall 3: SP-Berechnung mit verringerter LED-Anzahl und LED-Berechnung durch parallele Pixelorder (nach [Jos20])

Im Beispiel sind zwei Konstellationen im Vergleich. Die blauen und die roten Balken gehören zusammen ( $N_{\text{C}} = 1536 \wedge N_{\text{SP}} = 6321$ ), sowie analog die gelben und violetten Balken ( $N_{\text{C}} = 2400 \wedge N_{\text{SP}} = 9831$ ). Insbesondere ist die zweite Konstellation eine sehr anwendungsnahe und real geforderte Spezifizierung künftiger Local-Dimming Systeme für eine hohe LED-Anzahl, da mit zunehmender BLU-Auflösung auch eine hohe SP- und Condenserauflösung sinnvoll ist, um die hohe LED-Anzahl auch ausnutzen zu können. Daraus geht hervor, dass zukünftig die SP-Berechnung mit verringerter LED-Anzahl durchgeführt werden muss. Für eine SP-Anzahl von 6321 kann jedoch auf die Parallelisierung der Pixelorder verzichtet werden, dabei wird der gelbe Balken zwar doppelt so hoch, spart dadurch aber die für die Parallelisierung notwendige Hardware (Fall 3  $\rightarrow$  Fall 2). Insgesamt wäre für diese Konstellation also noch für eine Bildwiederholfrequenz von 60 Hz noch genug zeitlicher Puffer übrig. In Abbildung 6.37a ist ein HMI-Testbild gezeigt, zu dem für steigende LED-Zahlen das Powersaving mit dem hier entwickelten Algorithmus simuliert wurde. In Abbildung 6.37b ist das berechnete Powersaving gegen die steigende LED-Zahl aufgetragen und zeigt die

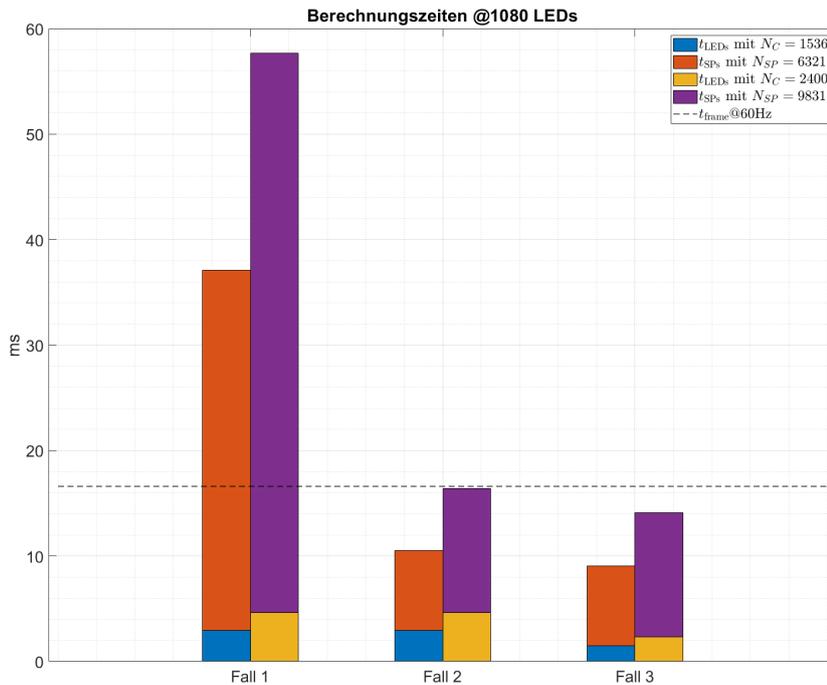
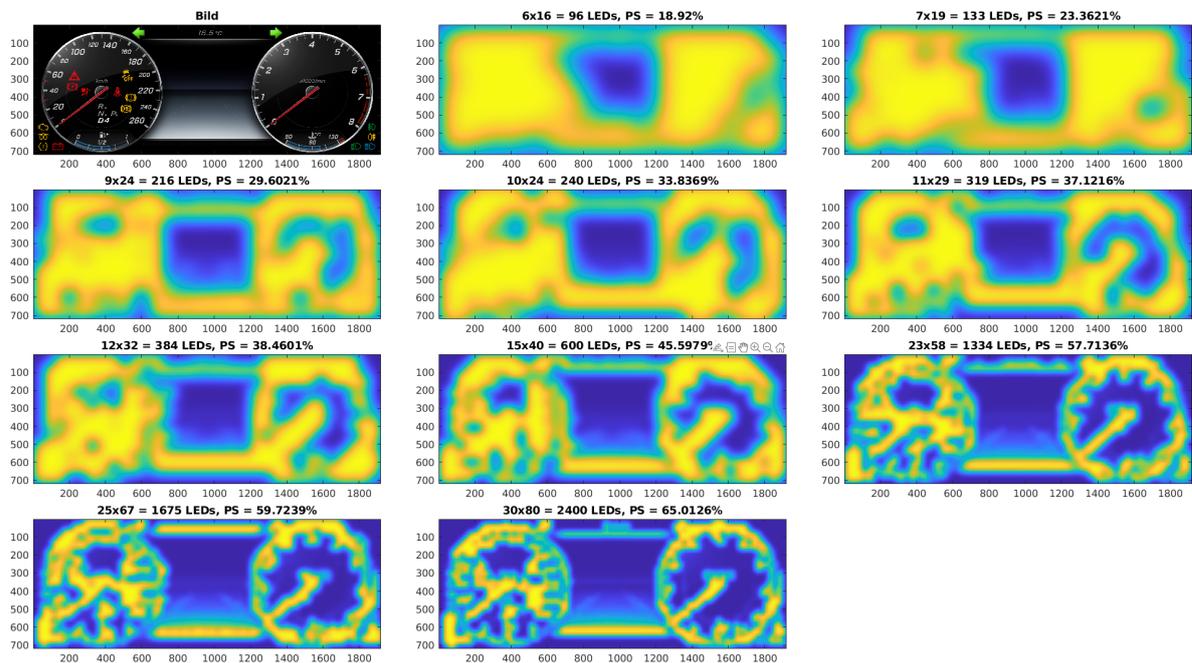
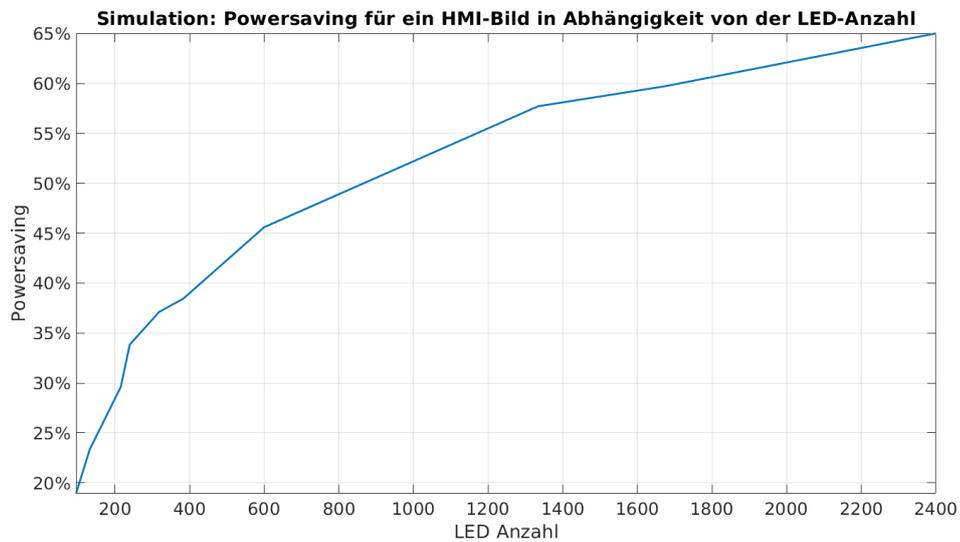


Abbildung 6.36.: Berechnungszeiten bei 1080 LEDs mit Optimierungsmethoden mit einer Systemclock mit  $f_s = 200$  MHz.

weitere Leistungseinsparung. Die Kurve wird jedoch erwartungsgemäß immer flacher, sodass eine zu hohe LED-Anzahl keinen Mehrwert mehr bietet und der Hardwarebedarf für das IP wie oben erwähnt ansteigt. Einen guten Kompromiss zwischen der Local-Dimming-Performance und Kosten stellen BLUs bis  $\approx 1000$  LEDs dar. Mit solch einer BLU kann ein Trade-Off zwischen Kosten und Nutzen hergestellt werden. Im nächsten Abschnitt wird das entwickelte Local-Dimming System als Ganzes betrachtet und ein Ausblick auf zukünftige Entwicklungen gegeben.



(a) Backlight und zugehöriges Testbild.



(b) Powersaving aufgetragen gegen die Anzahl der Dimming Zonen  $N_L$  in der BLU.

Abbildung 6.37.: Simuliertes Powersaving mit Visualisierung des Backlights für ein HMI und steigenden BLU-Auflösungen.



welcher dann den Defekt unter geringen Einbußen des Powersavings weitestgehend kompensieren kann. Ein weiterer Vorteil der Modellierungsmethode für die Nutzung im Automobil ist die Reduktion der benötigten LSF-Messungen, was der Anzahl der Modelle zuzüglich einer All-On Messung ( $N'_L + 1$ ) entspricht. Zusammengefasst die USPs im Überblick:

- Integration auf der Pixelpipeline lässt eine effiziente Implementierung im TCON zu
- Beschleunigte Optimierung und Pixelkompensation ermöglichen Videobetrieb für hohe LED-Auflösungen
- Neuartiges LSF Modell reduziert die Komplexität bei Berücksichtigung des Crosstalks
- Qualitätserhaltende und verbessernde Methoden heben die Wertigkeit des Displays an

Diese Features unterscheiden sich substantiell vom SoA und stellen eine adaptierbare Lösung für automotiv Displays dar.

### Erhöhte Bildqualität

In Abbildung 7.2 ist eine Leuchtdichte-Messung eines Bildes, welches auf einem Prototyp dargestellt wird, gezeigt. Um die wenigen Demonstratoren vor Überlastung zu schützen, wird die Leuchtdichte über die StromEinstellung der LED Treiber reduziert. Sie kann jedoch einfach durch Erhöhung des Diodenstromes hoch-skaliert werden. In der genannten Abbildung links ist das ursprüngliche, ungedimmte Bild abgebildet, während rechts die Messung die Leuchtdichteverteilung anzeigt, welche aus der Anwendung des Local-Dimmings resultiert. Es ist gut zu erkennen, dass die hohen Leuchtdichten sehr gut durch das Dimming getroffen werden. Für den unteren Bereich ist zur besseren Darstellung die Skalierung der unteren Bilder auf den Bereich zwischen  $0 \text{ cd m}^{-2}$  und  $2 \text{ cd m}^{-2}$  begrenzt. Dadurch wird die hohe Leuchtdichteverbesserung der dunklen Bereiche, insbesondere in den Displayecken verdeutlicht. Da der Schwarzwert sehr wichtig für die Wahrnehmung und visuelle Qualität ist ([Tan+18; HSX16b]), sorgt der entwickelte Local-Dimming Algorithmus für einen signifikanten Mehrwert und sichtbare Steigerung der Bildqualität.

Aufgrund der Berücksichtigung des Crosstalks zwischen den LEDs, wird die gewünschte Peak-Leuchtdichte

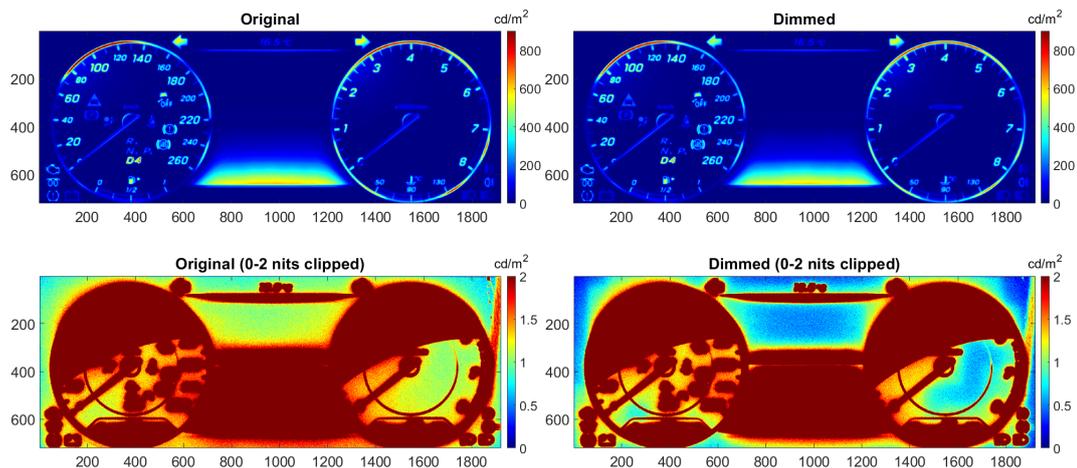


Abbildung 7.2.: Leuchtdichteaufnahme eines Prototyps. Die unteren Bilder sind zur Verdeutlichung bei  $2 \text{ cd m}^{-2}$  abgeschnitten und der Wertebereich darauf skaliert. Links: Original. Rechts: mit Local-Dimming. Bildteile in [SAX20] veröffentlicht.

gut getroffen. Das wird auch in Abbildung 7.3 im Vergleich zu einem SoA-Ansatz, bei welchem die LED-Werte durch das *Downsampling* des Bildinhalts auf die Matrix-Backlight Auflösung bestimmt werden (siehe Abschnitt 3.5.2), deutlich. Der SoA liefert eine viel zu geringe Leuchtdichte. Dieses Verhalten wird mit dem neuen Local-Dimming Prozessor aus dieser Arbeit systeminhärent unterdrückt und die benötigte Leuchtdichte gewährleistet.

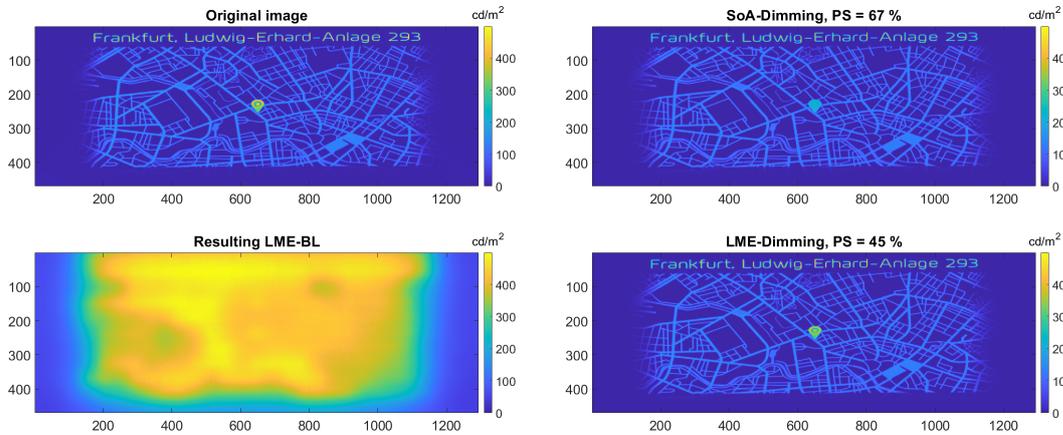


Abbildung 7.3.: Leuchtdichteaufnahmen eines Testbildes im Vergleich zwischen einem SoA Local-Dimming Algorithmus und der neuen IP. Einheit ist  $\text{cd m}^{-2}$ . Das Clipping im SoA Fall ist deutlich erkennbar.

### Hardwarekomplexität des Designs

Bei dem für die Direct-Lit Prototypen genutzt FPGA handelt sich um ein *Xilinx Kintex 7 XC7K325T* in einem *FFG900* Ball Grid Array (BGA, Kugelgitteranordnung) Gehäuse. Für die gesamte Implementierung inklusive *UART* und *SPI* wird für die Realisierung 1 MBit an SRAM benötigt. Dieser wird im FPGA durch Block-RAM bereitgestellt. Dieser Bedarf ist in Tabelle 7.1 detailliert aufgeführt.

Zusätzlich werden für die sequentielle Logik mit Pipelining 13 % der vorhandenen FFs genutzt. Von den verfügbaren 203800 6 Bit-LUTs ([Xil18]) werden für die Realisierung der Logik 21,65 % verwendet. Da es sich hier um ein FPGA handelt, ist zu erwähnen, dass die genutzten Multiplizierer tatsächlich durch die Logik realisiert und *keine* im FPGA selbst verfügbaren Hardwaremultiplizierer verwendet wurden. Der Grund hierfür ist, dass eine Logikrealisierung flexibel für verschiedene FPGAs synthetisiert werden kann, während die fest verfügbaren Hardwarekomponenten eines FPGAs an eine entsprechende FPGA Modellreihe gebunden ist. Zudem kann das Design einfacher in ein ASIC wie einen Timing-Controller (TCON) überführt werden. Das während dieser Arbeit erzeugte IP, welche durch Verilog beschrieben wurde, kann somit flexibel auf verschiedenen Hardware (FPGAs, ASICs) eingesetzt werden.

### Reduktion des Leistungsbedarfs

Neben der moderaten Hardwarekosten spielt auch das Powersaving eine wichtige Rolle. Dabei kann das Powersaving, je nach Vorgabe variieren, da eine bestimmte Parameterwahl entscheidend für die resultierende Einsparung ist. Diese besagten Parameter haben jedoch auch Einfluss auf die visuelle Qualität wie beispielsweise die Halo-Unterdrückung. In Tabelle 7.2 wird dazu eine Übersicht bei verschiedenen Parametersätzen gezeigt. Zugrunde liegen die Testbilder aus Anhang B. Der räumliche Faktor ist nach Formel (6.1) in

Bezeichnung	Adressbreite (in Bit)	tatsächliche An- zahl Adressen	Wortbreite (in Bit)	Gesamt (in kBit)
3 × Gamma_ROM	10	1024	20	3 × 20,48
3 × Degamma_ROM	11	2048	10	3 × 20,48
2 × Condenser_RAM	11	1536	10	2 × 15,36
2 × Initialwerte_RAM	8	240	8	2 × 1,92
2 × Sample-Point_RAM	13	6321	16	2 × 101,136
CondAddr2LEDnum_ROM	11	1536	8	12,288
Pixelorder_ROM	11	1536	152	233,472
LED-Buffer_RAM	8	240	16	3,856
Modelle_ROMs	-	-	-	454,775
			Summe:	1064,103

Tabelle 7.1.: Zusammenfassende Übersicht über den SRAM Bedarf des Local-Dimming Prozessors. In Modelle\_ROMs werden alle Modelle sowie dazugehörige LUTs zusammengefasst. Dazu gehören die entsprechenden Mappings sowie die eigentlichen Modelle, welche 367,308 kBit davon in Anspruch nehmen.

Abschnitt 6.2.2.1 eingeführt. Die minimalen LED-Werte wurden in Abschnitt 6.2.2.2 behandelt. Die Anzahl der inkrementierbaren LEDs (siehe auch Abschnitt 6.3.2) bieten einen weiteren Hebel um das Powersaving auf Kosten der erreichbaren Leuchtdichte zu erhöhen.

In Abbildung 7.4 (a) wird ein Beispiel eines HMI Bildes mit zugehörigen Condenserwerten und berechneten LED-Werten gezeigt. In (b) wird das auf den Modellen basierende Backlight in Sample-Point Auflösung dargestellt. Dieses dient der Pixelkompensation. Zu erkennen sind die Abdunklungen der Ecken und der oberen schwarzen Fläche. Dadurch wird der Kontrast deutlich verbessert und das Display wirkt hochwertiger. Es wird ein Powersaving von 46 % erreicht.

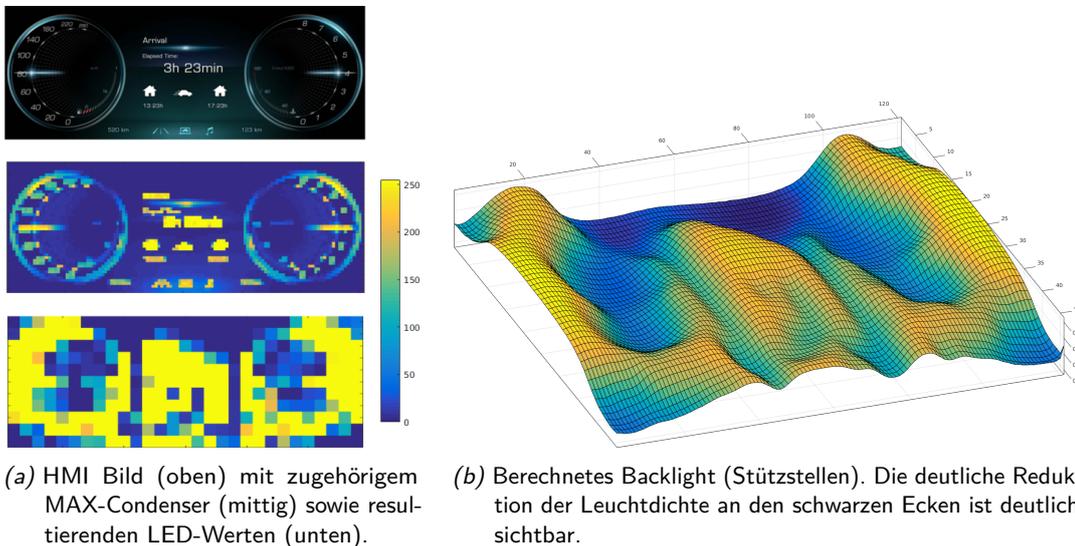


Abbildung 7.4.: Visualisierung des Condenserinhaltes und resultierender LED-Werte (8 Bit) und zugehöriges Backlight.

In Kombination mit Boosting (Abschnitt 6.1.2) kann jedoch eine noch lokalere Leuchtdichteverteilung erreicht werden. Dieser Parameter hat einen hohen Einfluss auf das Powersaving Ergebnis. Die Ursache

hierfür ist die LSF in Kombination mit dem Bildinhalt. Wird die Zielleuchtdichte von einer LED nicht erreicht, so wird das Powersaving zunehmend kleiner bei einer höher werdenden Anzahl an inkrementierbaren LEDs pro Condensierzelle, da ihre Einflüsse auf diese Zelle geringer sind. Wird der Boosting-Faktor so gewählt, dass die Zielleuchtdichte mit einer LED erreicht werden kann, liegt die erreichte Leuchtdichte am Optimum.

Parametersatz				
Räumlicher Faktor $f_{r,b}$	LED- $v_{\min}$	LED-Min-Wert	erlaubte Anzahl inkrementierbarer LEDs pro Condensierzelle	$\phi$ Powersaving
$\infty$	0 %		1	<b>54,97 %</b>
6,25	0 %		1	<b>51,98 %</b>
6,25	7,8 %		1	<b>50,79 %</b>
$\infty$	0 %		3	<b>43,31 %</b>
6,25	0 %		3	<b>41,12 %</b>
6,25	7,8 %		3	<b>40,13 %</b>
$\infty$	0 %		5	<b>36,49 %</b>
6,25	0 %		5	<b>34,92 %</b>
6,25	7,8 %		5	<b>34,09 %</b>

Tabelle 7.2.: Mittleres Powersaving der Testbilder aus Anhang B für verschiedene Parametersätze der 240er BLU.

**Einfluss der LSF**

Die aus den vorangegangenen Kapiteln gewonnenen Erkenntnisse bzgl. der Auswirkungen der LSF auf die Performance ist in Abbildung 7.5 dargestellt. Eine höhere Lokalität führt somit zu einer höheren Effizienz, also zu einem höheren Powersaving. Jedoch steigt die Anfälligkeit von *Halo* Artefakten an. Auf der anderen Seite liefert eine *globalere* LSF eine höhere Uniformität des Backlights und eine höhere *Robustheit* gegenüber Produktionsschwankungen, jedoch auch eine schlechtere Effizienz und keine hohe Kontrastverbesserung. Eine höhere Lokalität kann neben dem optischen Design durch Nutzung einer höheren LED-Anzahl erreicht werden.

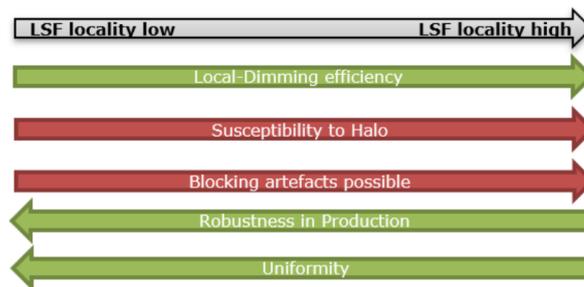


Abbildung 7.5.: Einfluss der LSF Lokalität. Entnommen aus [SAX20].

## Schlussfolgerung

Zusammenfassend können die Einflussfaktoren, welche die Performance des Local-Dimming Systems beeinflussen, in Abbildung 7.6 resümiert werden. Die Hardwarekosten müssen zusammen mit der LSF-Form und dem Algorithmus in Einklang gebracht werden. In einer realen Anwendung ist der Kostendruck sehr hoch. Bei gutem Trade-Off entsteht jedoch ein Displaysystem mit einer hohen Wertigkeit, welches vergleichbare Bildqualität wie ein OLED-Display, bei gleichzeitig deutlich geringeren Kosten liefert.

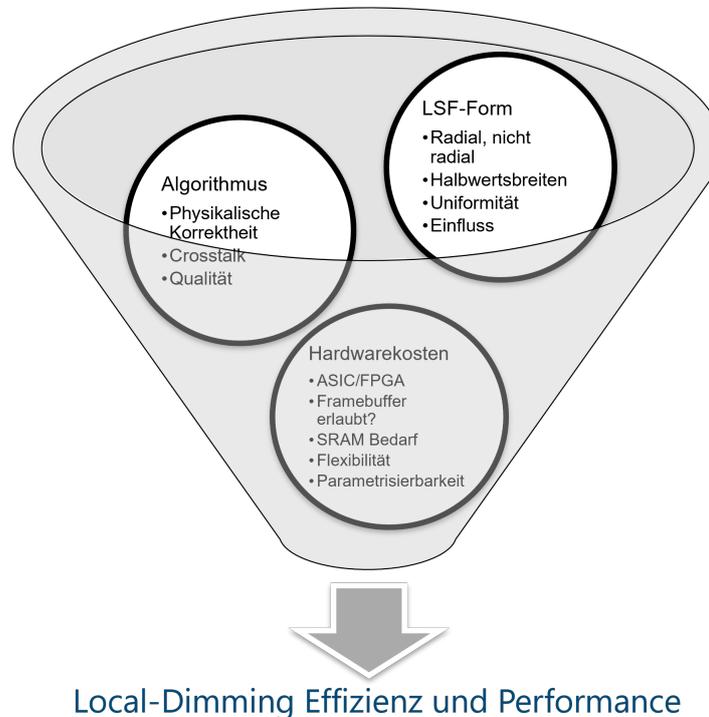


Abbildung 7.6.: Einflussfaktoren auf die Local-Dimming Effizienz und Performance.

Insgesamt bietet der hier vorgestellte Algorithmus viel Flexibilität bei der Parametrisierung an spezielle Kundenwünsche und bei der Findung des Trade-Offs zwischen Powersaving und Unterdrückung von Artefakten. Die Kosten des Systems bleiben durch die LSF-Modellierungsmethode auch für weiter steigende LED-Auflösungen im Verhältnis moderat. Das Local-Dimming Hardware-Design lässt sich somit gut in ASICs realisieren wie beispielsweise beim Design eines TCONs. Da auf einen Frame-Buffer gänzlich verzichtet wird und der Local-Dimming Prozessor auf der Pixelpipeline arbeitet, ist der Speicherbedarf gering und ein *Einfrieren* des Bildinhaltes (ursächlich durch den Local-Dimming Prozessor) ist somit ausgeschlossen. Daneben sind Powersavingraten sowie die Verbesserungen beim Halo hoch. Damit steigert solch ein Local-Dimming System die Wertigkeit des Displays enorm und stellt eine Alternative zu OLED-Displays dar, da der Schwarzwert durch Local-Dimming deutlich verbessert wird.

Die Tauglichkeit und Einsatz im Automobil ist somit gerechtfertigt. In Abschnitt 6.5 wurden weiterhin Methoden vorgestellt, welche die Berechnungszeit reduzieren. Somit wird eine noch höhere BLU-Auflösung ermöglicht und erlaubt die künftige Anwendung des in dieser Arbeit entwickelten Algorithmus. Zusammen mit der Funktion gegen Ausfall einzelner LEDs können höhere ASIL Einstufungen ermöglicht werden. Das Local-Dimming-System kann im Automobil breit eingesetzt werden, wie beispielsweise für ICDs, CIDs

und auch Head-up-Displays (HUDs) mit Direct-Lit-Backlight. Auch eine Anwendung bei LC-basierenden Scheinwerfersystemen ist denkbar [Ben19]. ICD Prototypen mit dem hier vorgestellten Algorithmus wurden von mehreren OEMs getestet. Auch die Verwendung der hier entwickelten Local-Dimming Technologie in einem *Shy Tech* Display Demonstrator von *Continental* (siehe Kapitel 8) unterstreicht die Einsetzbarkeit im Automobil.

## 7.2. Weitere Arbeiten und Ausblick

### LCD-Trend im Automobil

Der anhaltende Trend zur Abkehr von Automobilen mit Verbrennungsmotor ist im Gange und wird aufgrund der Gesetzgebung noch weiter Fahrt aufnehmen. Auch das autonome Fahren entwickelt sich stetig voran. Der Wunsch der OEMs neue Funktionalitäten im Fahrzeug zu haben wächst somit weiter um sich gegen neue Konkurrenz behaupten zu können. Das über viele Jahrzehnte erarbeitete Know-How des Verbrennungsmotors und Komponenten wird immer weniger relevant. Neue Firmen drängen in den Automobilssektor, sei es als Zulieferer oder OEM. Beispielsweise zeigt *Sony* Ambitionen mit dem *Sony Vision-S 02* [Hub22].

Der Trend der Displays im Automobil geht in die Richtung *größer, breiter, hochwertiger*. Auch die Anzahl der Displays steigt. Es werden

- höhere Pixelauflösungen
- sowie mehr LEDs in der BLU

notwendig um dem Trend gerecht zu werden. Die Bildqualität (e.g. Kontrast/Schwarzwert) nimmt einen immer höheren Stellenwert ein. Durch eine höhere LED-Dichte in der BLU und einem adäquaten Local-Dimming System, kann die Bildqualität unter Beachtung der begrenzten Akkukapazität in E-Fahrzeugen weiter verbessert werden. Jede Komponente im Auto muss künftig noch weiter optimiert werden, um den Leistungsumsatz zu reduzieren.

In Abbildung 7.7 ist ein Display der Firma *China Star Optoelectronics Technology* abgebildet. Die BLU besteht aus 5096 Dimming Zonen und demonstriert den beschriebenen Trend. Auch die geringe Bautiefe zur Integration ins Interieur Design wird immer wichtiger. Ein solches Display soll für *Future Cars* bei einem OEM eingesetzt worden sein.



Abbildung 7.7.: Display mit einer Auflösung von  $1080 \times 7680$  Pixel und einer BLU mit 5096 Dimming Zonen. Hergestellt von *China Star Optoelectronics Technology*. Entnommen aus [Bla21].

Um die höhere BLU Auflösung, im Speziellen die LED-Dichte weiter zu steigern, bei gleichzeitiger Reduktion der Bautiefe, können die sogenannten *Mini-LEDs* eingesetzt werden. Auch die Uniformität wird durch

deren Einsatz weiter verbessert [Kik+19]. Diese grenzen sich zu den *Micro-LEDs* durch ihre Größe ab. Als *Micro-LEDs* werden die LEDs unter  $80\ \mu\text{m}$  bezeichnet. Als *Mini-LEDs* gelten die LEDs im Bereich zwischen  $80\ \mu\text{m}$  und  $500\ \mu\text{m}$  [Tan+18; Zhe+19a].

Die BLU, welche im Display aus Abbildung 7.7 verwendet wird, ist so eine *Mini-LED BLU* mit 5096 Dimming-Zonen, welche wiederum aus mehreren *Mini-LEDs* aufgebaut werden.

In [Mas+19] wird für ein 15,6 Zoll Display eine BLU aus 2048 Dimmingzonen vorgestellt, welche jeweils aus 9 LEDs der Dimension  $130\ \mu\text{m} \times 240\ \mu\text{m}$  aufgebaut sind. Der Halo Effekt wird durch die höhere LED-Anzahl weiter reduziert und der Kontrast erhöht [Zhe+19b; Xia+19; Hua+19]. In [Xia+19] wird dies mit einer BLU mit über 5000 Zones bei einem 75 Zoll Display gezeigt. Eine Zone besteht hier aus 4 LEDs. Um die immer weiter steigende LED-Anzahl in der BLU ansteuern zu können, reicht die in Abschnitt 3.5.3 vorgestellte *Direct-Drive Methode* nicht mehr aus, sodass es die LED-Matrix passiv oder aktiv angesteuert werden muss [Mas+19; Liu+19; Zhe+19b; Tin+19; Xia+19]. Die *Aktiv-Matrix Ansteuerung* liefert generell weniger Artefakte als die *Passiv-Matrix* (*Ghosting*, *Crosstalk*), ist jedoch in der Produktion deutlich teuer [Tin+19]. Ein weiterer Vorteil ist der geringere Spitzenstrom sowie eine geringere LED-Chipfläche (siehe Abbildung 7.8).

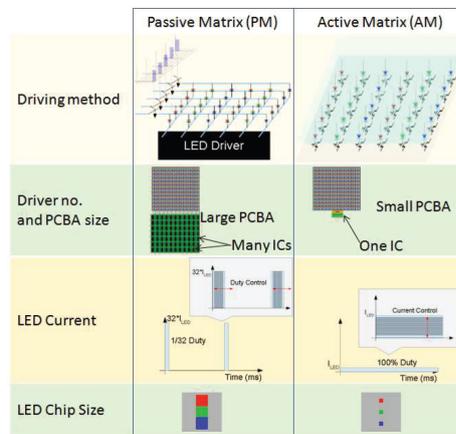


Abbildung 7.8.: Vergleich zwischen AM und PM Ansteuerung der BLU. Entnommen aus [Tin+19].

Um jedoch eine günstigere *Passiv-Matrix Ansteuerung* zu realisieren und Nachteile der *Passiv-Matrix* auszugleichen, sodass eine höhere Zeilenanzahl verwendet werden kann, kann das sogenannten *Consecutive Multiline Addressing* Schema von Xu u. a. [Xu+08] verwendet werden. Die Kombination von *Local-Dimming* mit *Multiline-Adressing-Ansteuerung* für sehr hoch auflösende BLUs ist ein herausforderndes Forschungsthema, welches Potential in sich birgt.

### Weitere Trends

Der allgemeine Trend des maschinellen Lernens kommt auch im Displaybereich an. Auch Versuche mithilfe künstlicher neuronaler Netze und *Local-Dimming* werden publiziert. In [Jo+18] wird ein *Convolutional Neural Network (CNN)* genutzt, um die *Pixelkompensation* mit dem *CNN* durchzuführen. Da die Ergebnisse sich schon bei kleinsten Änderungen stark unterscheiden können, ist diese Methode für den automotiven Einsatz nicht geeignet. Zudem müssen in diesem Fall die LED-Werte weiterhin gegeben sein. In [Jo+18] wurden die *Postgamma APL* als LED-Werte angenommen, welche für das Erreichen der ursprünglichen Leuchtdichte in der Regel zu gering sind. Der Speicherbedarf ist zudem sehr hoch, da mehrere Layer in

immer größerer Auflösung gespeichert werden müssen.

Der Wunsch nach noch höheren Kontrasten und HDR führt zur Entwicklung von sogenanntem Pixel-Dimming [Che+19; Che+17c]. Dabei wird der Aufbau des LCDs durch ein zweites, geringer aufgelöstes LC-Panel ohne Farbfilter erweitert. Dieses zweite Panel im optischen Stack kann zur weiteren Modulation des Backlights genutzt werden und die Kontraste weiter erhöhen. Dabei wird eine Farbtiefe von bis zu 14 Bit erreicht und ein Kontrastverhältnis von 5000:1 bis 200000:1. Der Vorteil gegenüber OLED-Display ergibt sich aus der Peak-Leuchtdichte und höherer Graustufe, welche mit OLED nicht erreicht wird.

In Abbildung 7.9 wird der Aufbau und ein Nachteil dieses *Dual-Stacks* verdeutlicht. Durch die ähnliche Gitterstruktur und leichter Falschalignment (*Misalignment*) wird der Moiré-Effekt schnell sichtbar und reduziert die optische Performance sehr stark. In [Che+17c] wird ein *PDSF* (*polarization dependent scattering film*) genutzt, um den Effekt zu verringern (siehe Abbildung 7.9b). Die passgenaue Ausrichtung der Panels ist ein Problem. Neben dem Moiré-Effekt stehen dem hohen Kontrast die reduzierte Effizienz und steigende Kosten gegenüber.

Insgesamt steht Pixel-Dimming noch am Anfang der Möglichkeiten hat jedoch mittelfristig für den automotiven Bereich keine Relevanz aufgrund der genannten Nachteile. Jedoch ist eine Anwendung im medizinischen Bereich möglich. Diese Technik bietet einen zusätzlichen Freiheitsgrad nämlich, dass sich die gesamte Transmission ein Produkt der Transmissionen beider LC-Panels ergibt. Daraus bietet sie in Zusammenhang mit Local-Dimming Möglichkeiten für die Optimierung weiterer Größen wie Energie-Effizienz, Kostenreduktion, und andere. Die Nutzung einer Direct-Lit BLU mit Local-Dimming anstelle heutiger Edge-Lits soll die Energie-Effizienz des Displaysystems signifikant steigern und zudem die Peak-Leuchtdichte um ein Mehrfaches erhöhen können. Gerade für HDR ist diese Peak-Leuchtdichte ein Schlüsselparameter.

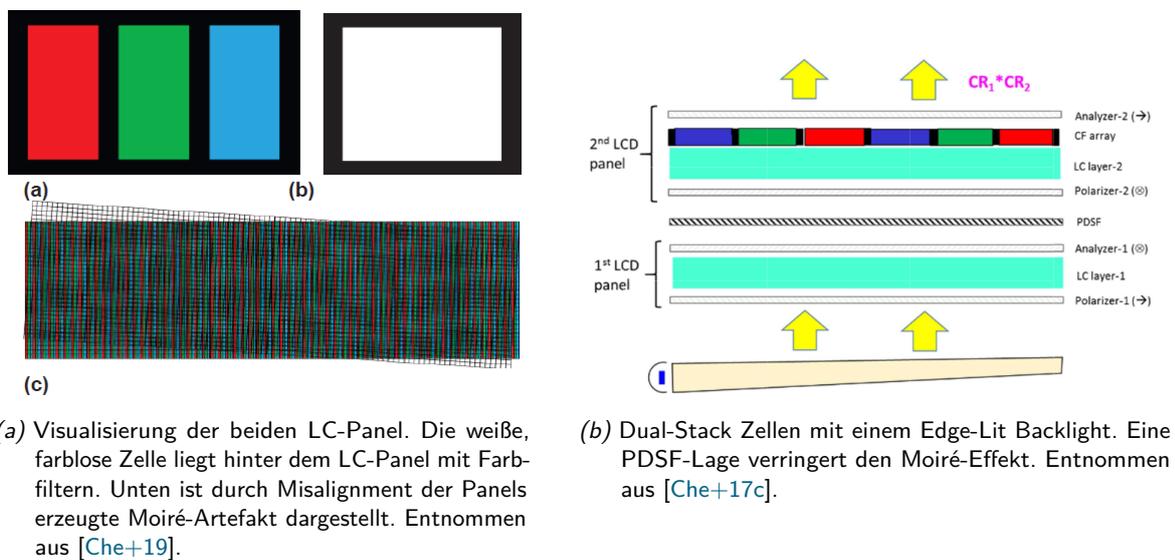


Abbildung 7.9.: Funktionsprinzip von Pixeldimming mit Dual-Stack LCD und mögliches Moiré Artefakt.

### Aufbauende Arbeiten

In den nächsten Jahren werden automotive LCDs zunehmend mit Matrix-BLUs ausgestattet, um Displays mit höherer Leuchtdichte, Kontrasten und Powersavingraten aufzuwerten. Basierend auf dieser Arbeit kann

der Local-Dimming Algorithmus mit den hier vorgestellten Methoden an noch höhere LED-Auflösungen angepasst werden. Zudem werden künftig mehr blaue LEDs mit Remote-Phosphor Color-Conversion (im Diffusor) [KSK18] eingesetzt um die optische Distanz zu reduzieren (vgl. Abbildung 2.14, Seite 24). Die BLU kann damit dünner konstruiert werden. Beispielsweise nutzt *Apple* solch eine BLU im Apple Pro XDR Display [Gu+20]. Ein Nachteil des Remote-Phosphors ist jedoch der entstehende Gelbshift [Gu+20]. Um diesen zu reduzieren, können im Local-Dimming Algorithmus Methoden integriert werden, welche diesem Drift algorithmisch entgegenwirken können. Beispielsweise durch die Limitierung der Parametersettings oder auch durch ein erweitertes LSF Modell, welches neben der Leuchtdichte bzw. Einfluss auch den Farbort modelliert. Durch einen erweiterten Condenser mit Luminanz und Farbörter können die Werte als Optimierungsziel für den Algorithmus genutzt und in der Optimierungsphase beachtet werden.

LCDs werden die Mainstream-Displaytechnologie bleiben und zwar auch für automotive Anwendung [XAS22]. Mittelfristig wird ein hoher Anteil von Local-Dimming-LCDs erwartet (vgl. Abschnitt 3.2). Innovative Displays und Displaykonfigurationen- und Systeme wie *Shy Tech* (siehe Abbildung 8.1, Seite 241) oder Projektionsdisplays wie HUDs sind in absehbarer Zeit auf dem Markt zu erwarten. Local-Dimming Algorithmen und optimale Matrix-BLUs sind wichtige Enabler für hohe Energie-Effizienz und höhere Bildqualität. Sie werden ein zukunftssträchtiges Forschungsthema bleiben.

## 8. Zusammenfassung

Im Rahmen dieser Arbeit mit dem Titel „Effizienter Local-Dimming Algorithmus und Matrix-Backlight-Designoptimierung für LC-Displays zur Steigerung der Energie-Effizienz und der Bildqualität mit Fokus auf automotive-spezifische Anforderungen“, wurde eine neue Local-Dimming Technologie unter Anwendung neuer Methoden entwickelt und Backlight-Unit (BLU) Designparameter abgeleitet. Die beiden wichtigen Ziele der Steigerung der Energie-Effizienz und der Bildqualität wurden erreicht und die Displayperformance signifikant erhöht. Die Anwendbarkeit im Automobil wird zudem durch die Nutzung der hier entwickelten Technologie in einem *Shy Tech* Display (siehe Abbildung 8.1) unter Beweis gestellt.

Die Automotive-Anwendung stellt hohe und diffizile Anforderungen an Displaysysteme, die analysiert und berücksichtigt werden müssen. Als Beispiel seien HMI-Inhalte mit hohem Kontrast und kreisförmige Strukturen genannt. Wichtig ist das Treffen der Ziel-Leuchtdichte der kleinen hellen Flächen und dünnen Streifen. Die Fehler-Toleranz ist gering, da die Betrachter eine gewisse Erwartungshaltung gegenüber den bekannten künstlichen Inhalten haben. Die Automotive-Standards bzgl. Qualität und Sicherheiten müssen erfüllt werden. Nicht zuletzt ist es eine kosten-sensitive Anwendung.

Ein wesentlicher Kostentreiber ist das Matrix-Backlight, dessen Performance mit der LED-Anzahl und damit mit den Kosten ansteigt. Hier setzt die Arbeit an und optimiert das Design des Matrix-Backlight an den Local-Dimming-Algorithmus, sodass die Performance bei gleicher LED-Anzahl gesteigert werden kann (Kapitel 4).

Hierfür wurden verschiedene BLUs vermessen und untersucht. Eine Modellfunktion für die Light-Spread-Function (LSF) wurde erstellt, welche mit 3 Parametern die LSF über einen weiten Bereich beschreibt und so auch eine quantitative Aussage über Kontrast beim Local-Dimming ermöglicht. Die Aspekte des BLU-Designs wurden mithilfe des Modells und der optischen Simulationen ausgiebig untersucht und ihre Auswirkungen auf die Local-Dimming-Performance analysiert. Betrachtet wurden u. a. Homogenität, Robustheit in der Produktion und Lokalität einer LSF. Die Messgröße ist das Powersaving.

Basierend auf den gewonnenen Erkenntnissen wurde ein wichtiger Design-Parameter der BLU bzw. der LSF eingeführt. Dabei handelt es sich um das FWHM/LED-Pitch Verhältnis ( $r_{50\%}$ ) welches in Abschnitt 4.4 beschrieben wurde. Es wird ein  $r_{50\%} \geq 1,5$  bei einem maximalen Einfluss von  $\approx 30\%$  empfohlen, um für radial-invariante LSFs einen guten Trade-Off zwischen Effizienz, Bildqualität und Robustheit zu liefern. Auch  $r_{80\%}$  liefert eine hohe Korrelation zur Uniformität einer BLU und kann für die Spezifikation einer BLU verwendet werden.

Aufgrund der steigenden LED-Anzahl eines Matrix-Backlights muss der bisherige Ansatz des SSC Local-Dimming-Algorithmus, der am Lehrstuhl in Rahmen dreier Promotionen ([Sch17; Jun14; Alb10]) entstand und sich schwerpunktmäßig der TV-Anwendung mit Edge-Lit widmete, grundlegend erneuert werden. Der Speicherbedarf würde proportional mit der LED-Anzahl ansteigen. Ebenso steigt die Berechnungszeit, welche

bei einer höheren LED-Anzahl eine Frame-Periode (e. g. 16 ms) übersteigt. Die Grundidee des Greedy Prinzips wird im neuen Algorithmus als Merkmal beibehalten, da die Energie-Effizienz weiterhin im Fokus steht. Um die zeitliche Grenze der Frame-Dauer einzuhalten, wird der LED-Crosstalk untersucht und eine Methode hergeleitet, die sich bei der LED-Optimierung auf die einflussreichsten LEDs konzentriert. Die Rechenzeit für die LED-Optimierung bei  $N_L = 240$  LEDs sinkt auf die Größenordnung von  $\approx 18\%$  des Algorithmus zum Startpunkt dieser Arbeit, während der Crosstalk weiterhin berücksichtigt wird und die so errechnete Lösung nahe beim Optimum liegt.

Eine weitere Hürde ist die Rechenzeit für die Pixelkompensation, insbesondere die notwendige Berechnung der pixelgenauen Backlight-Verteilung. Die Methode der Betrachtung einer begrenzten Anzahl der LEDs wird hier auch verwendet. Während bei der LED-Berechnung 1% - 2% Abweichung vom Optimum problemlos akzeptiert wird, muss das Backlight genauer bestimmt werden, um visuelle Artefakte zu vermeiden. Folglich wird hierfür eine Korrektur-Funktion für die Berechnung des Backlights entwickelt, welche die Abweichung um einen Faktor 4 reduziert.

Die Berechnung und Optimierung sowie die Pixelkompensation verwenden ein in dieser Arbeit entwickeltes LSF-Modellkonzept (Abschnitt 5.3). Dadurch wurde der Speicherbedarf im Vergleich zu frühen Implementierungen drastisch verringert. Bei  $N_C = 1536$  Condenserezellen und  $N_L = 240$  LEDs beträgt der SRAM Bedarf der neuen Modelle und Pixelorder nur noch 7% des Bedarfs des SoAs zu Beginn dieser Arbeit. Bei selbiger Condenseranzahl und  $N_L = 1080$  beträgt der Bedarf der neuen Methoden nur noch 7% des SoAs.

Die entwickelte Systemarchitektur (Abschnitt 7.1) der IP ist parametrierbar und lässt sich gut auf höhere LED-Zahlen anpassen. Insgesamt sind die Kosten moderat. Für die Realisierung wird 1 MBit an SRAM benötigt. Die benötigte Logik entspricht etwa  $400 \cdot 10^3$  NAND-Gatter-Äquivalenten [Xu+18b].

Um den Mehrwert des Local-Dimmings weiter zu erhöhen, wird der Algorithmus um weitere Funktionen erweitert. Das Boosting erhöht die Lokalität des Backlights, sodass mindestens Powersaving oder Bildqualität oder beides verbessert wird. Das Halo Artefakt wird mittels lokal einstellbaren zweidimensionalen Filters entschärft. Zusätzlich wird der Schwarzwert vor allem an den Ecken verbessert (ein Faktor von  $> 6,3$  ist bei typischem HMI möglich), was zur Steigerung der Wertigkeit des Auto-Interieurs beiträgt (siehe auch [SAX20]). Der Algorithmus kann zudem den Ausfall einzelner LEDs weitgehend abfangen, sodass das System eine höhere ASIL-Stufe erreichen kann.

Je nach Wahl der Steuerungsparameter des Algorithmus, welche verschiedene Kompromisse abbilden und vom Anwender festgesetzt werden können, liegt das resultierende Powersaving (für eine 240 LED-BLU) für ein HMI Testset zwischen 34,09% und 54,94% (vgl. Tabelle 7.2 auf Seite 233).

Der entwickelte Algorithmus wurde anhand mehrerer BLUs mit bis zu 319 LEDs und verschiedenen LC-Panels validiert. Der erreichte Stand kann eine BLU mit 1080 LEDs und höhere Displayauflösungen abdecken, sodass er in den nächsten 10 Jahren mit den meisten BLUs anwendbar bleibt.

In dieser Arbeit wurde gezeigt, dass ein hocheffizientes Displaysystem bzw. ein sehr guter Performance/Kosten-Kompromiss realisiert werden kann, wenn BLU-Design und Algorithmus-Entwicklung aufeinander optimiert werden. Die neuen Methoden sorgen für eine exzellente Bildqualität bei hoher Effizienz. Dabei werden die hohen Automobil-Anforderungen wie das Erreichen hoher Peak-Leuchtdichte, exzellenter Schwarzwert,

Sicherheitsfunktionen etc. erfüllt. Ein solches Display kann somit für eine Differenzierung in Automobilen sorgen. Der Local-Dimming Algorithmus wurde in HDL codiert bzw. beschrieben und auf verschiedenen FPGAs implementiert. Mehrere Prototypen mit verschiedenen BLUs und LC-Panels wurden validiert. Neben den eigenen visuellen Tests am Lehrstuhl wurden die Prototypen auch von Projektpartnern validiert, die zudem eigenen Demonstratoren mit den im Rahmen dieser Arbeit entwickelten IPs aufbauten und sie der Öffentlichkeit wie auf Messen bzw. bei ihren Kunden präsentierten.

Die in dieser Arbeit entwickelte Technologie ermöglicht auch innovative Produktideen und Produktkonzepte. Ein gutes Beispiel ist die Verwendung in einem sogenannten *Shy Tech* Display, welches vom Projektpartner *Continental Automotive* entwickelt wurde [Hub21]. Abbildung 8.1 zeigt Fotografien eines Demonstrators. Es handelt sich um ein LCD, auf dessen Displayoberfläche mit spezieller Drucktechnik eine Holzoptik aufgetragen ist. Das im Rahmen dieser Arbeit entwickelte Local-Dimming IP wird in diesem Produkt zur Berechnung der LED-Werte und der Pixelkompensation genutzt. Die Local-Dimming-Technologie liefert die Ziel-Leuchtdichte für die hellen Bildbereiche, die durch die Holzoptik erscheinen. Zugleich sorgt sie dafür, dass die schwarzen Bereiche des Bildes aufgrund des signifikant verbesserten Schwarzwertes hinter der Holzoptik „verschwinden“. Das Display wird somit nahtlos im Armaturenbrett integriert (Abbildung 8.1) und kann im Automobil ein wertvolles Differenzierungsmerkmal darstellen. Die Einhaltung der hohen Leuchtdichte



(a) Display im ausgeschaltetem Zustand. Foto aus [Hub21].



(b) Display mit Entertainment Inhalt. Foto aus [Hub21].



(c) Display mit typischen Informationen für den Fahrzeugführer. Foto aus [Hub21].



(d) Seitliche Ansicht auf das Shy Tech Display. Foto aus [Kra21].

Abbildung 8.1.: Shy Tech Display von *Continental* mit aktivem Local-Dimming mit Holzdekor. Die Local-Dimming IP ist die in dieser Arbeit entwickelte Technologie.

und signifikant verbesserte Schwarzwerte sind bei diesem Produkt ausschlaggebend, die mit dem Local-Dimming-Algorithmus aus dieser Arbeit erreicht werden können. Außerdem wird das Problem der erhöhten Leistungsaufnahme aufgrund der benötigten höheren Leuchtdichte, die durch die Holzoptik durchscheinen muss, mit Local-Dimming entschärft. Laut *Continental* soll im Jahr 2023 dieses Display mit Local-Dimming bei einem OEM in Serie gehen. Es zeigt sich, dass mit dieser Arbeit die Local-Dimming-Technologie für den Einsatz im Automobil zur Marktreife gebracht und Design-In/Win gewonnen werden können. Dabei bilden die hohe Energie-Effizienz, hohe visuelle Qualität und effiziente Hardware zusammen das entscheidende Merkmal.

# A. Weitere Abbildungen

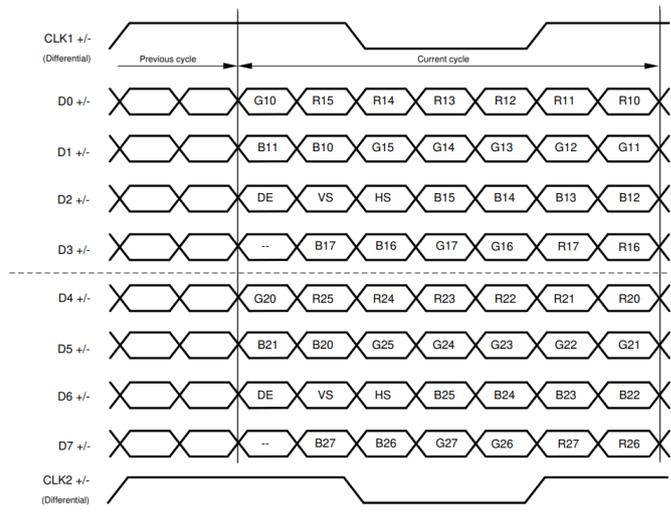


Abbildung A.1.: Mapping und Schema des FPD-Link mit 2 parallel übertragenen Kanälen. Entnommen aus [Tex20].

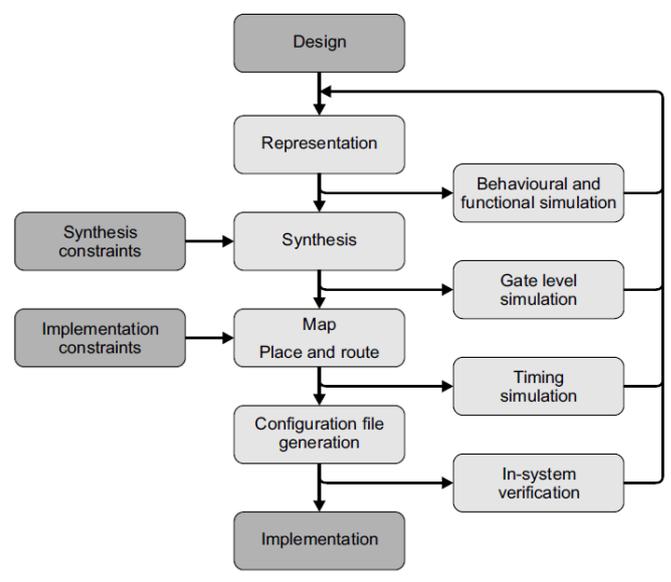
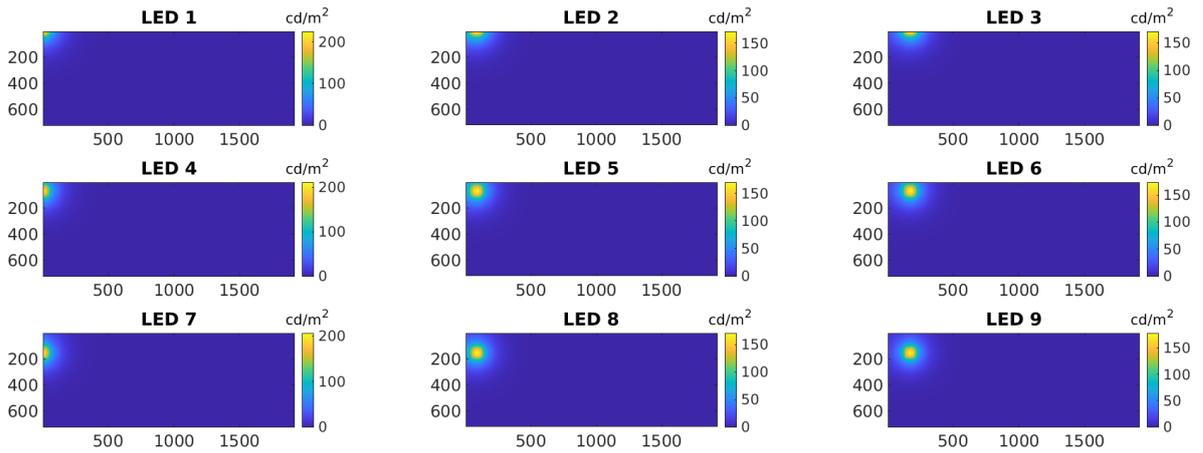
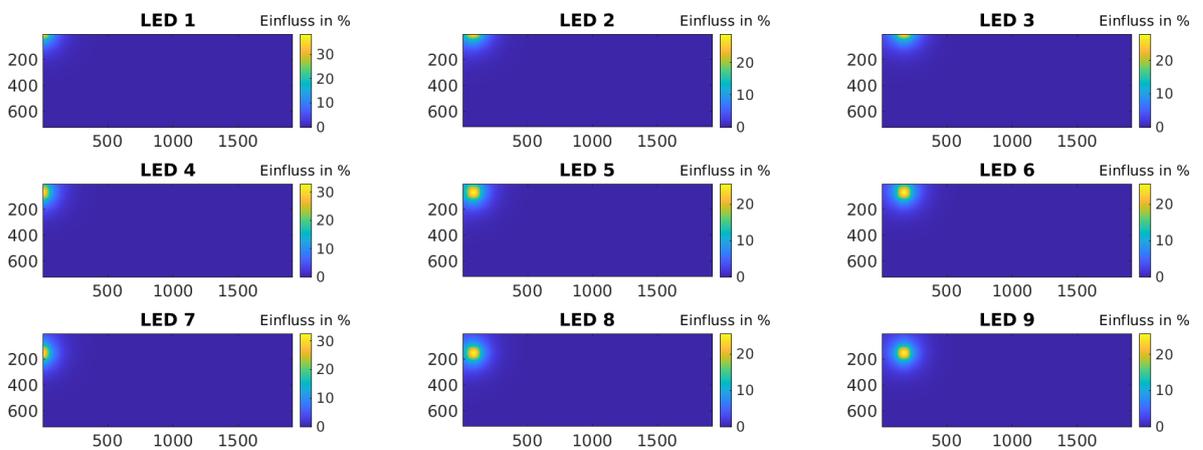


Abbildung A.2.: Ablauf einer Design-Implementierung für ein FPGA. Entnommen aus [Bai11].



(a) Absolute Leuchtdichteverteilung.



(b) Resultierender Einfluss der LEDs.

Abbildung A.3.: 9 Messungen der LSFs einer 240 BLU.

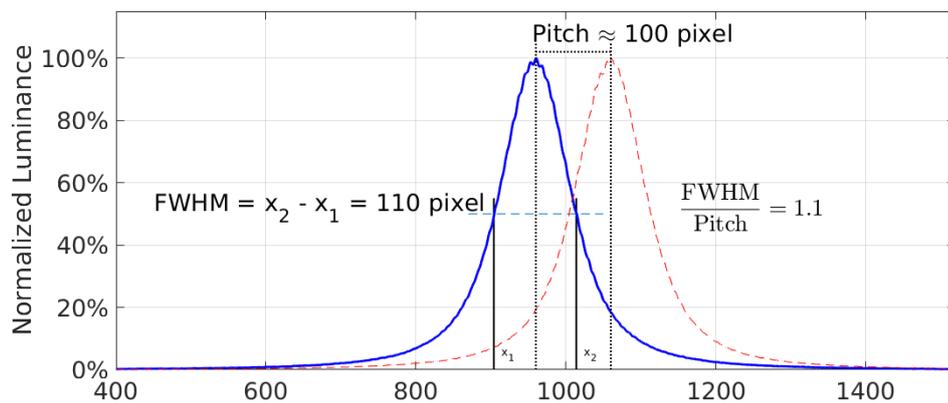


Abbildung A.4.:  $r_{50\%}$  der 133er BLU.

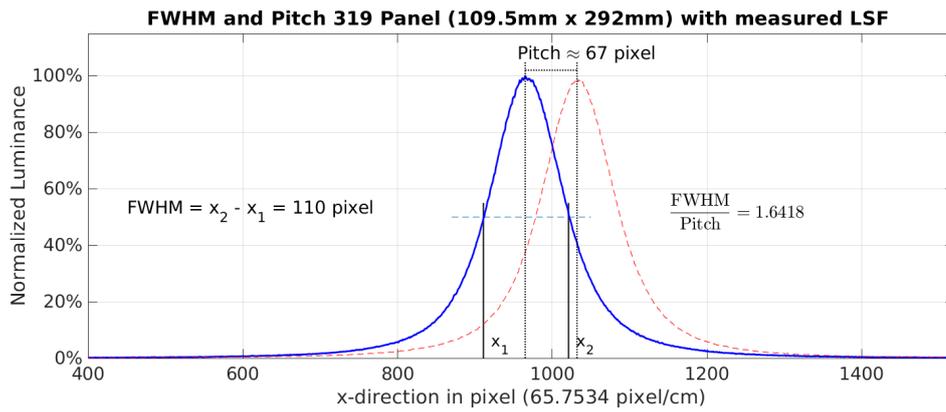


Abbildung A.5.:  $r_{50\%}$  der 319er BLU.

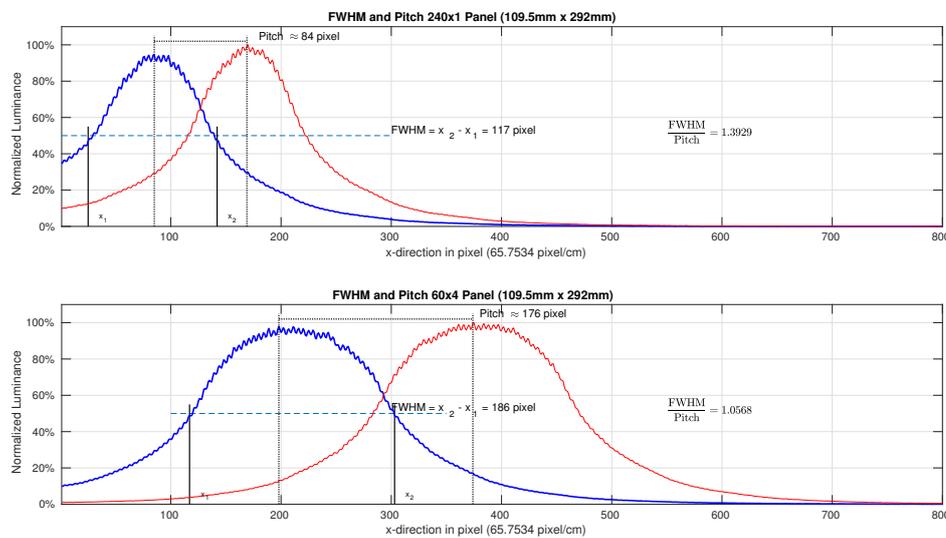


Abbildung A.6.:  $r_{50\%}$  der 240er BLU mit 240 bzw. 60 Zonen.

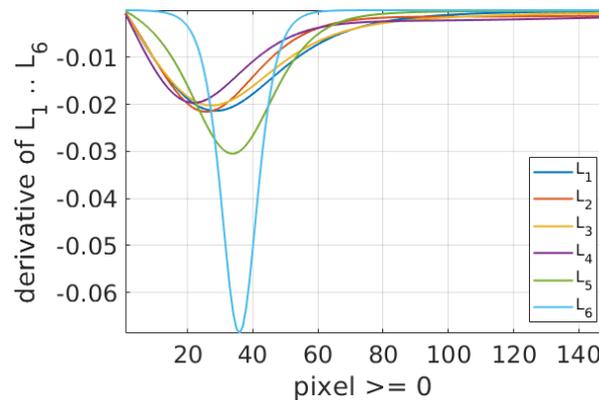


Abbildung A.7.: Steigung der 6 ausgewählten Modelle mit konstantem  $r_{50\%}$ .

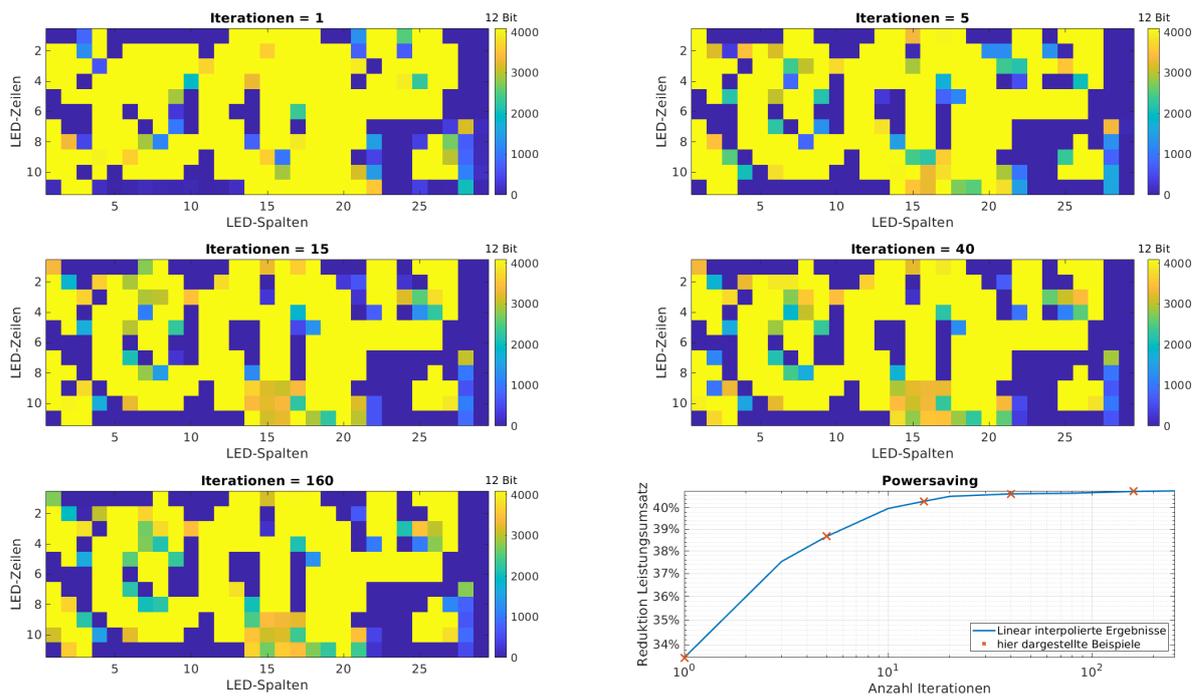


Abbildung A.8.: Final berechnete LED Werte bei verschiedenen Iterationszahlen und Leistungseinsparung für Bild 6 aus Abbildung 5.36, Seite 165.

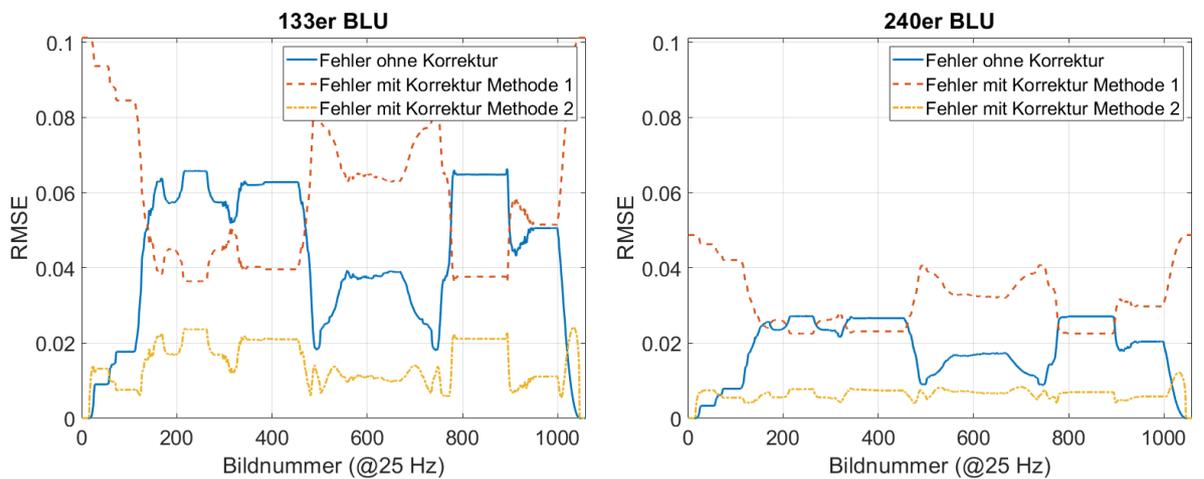


Abbildung A.9.: Resultierender RMSE des berechneten Backlights durch Betrachtung 7 x 7 LEDs mit Offset-Korrektur pro Sample-Point bei Testvideo 1.

## B. Übersicht Testbilder

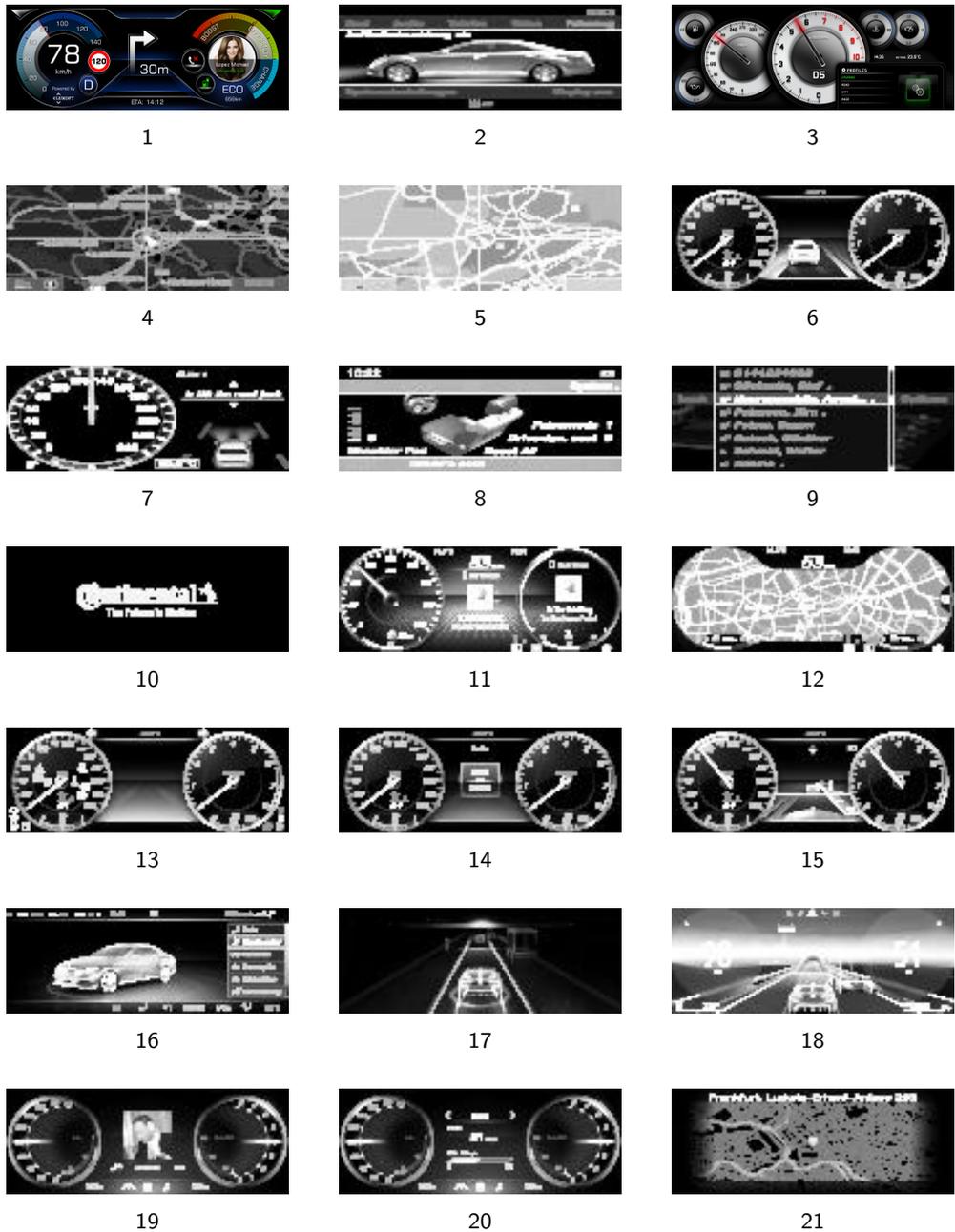
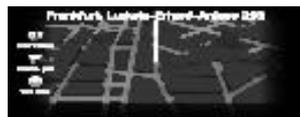


Abbildung B.1.: Testbilder mit 8:3 Seitenverhältnis. Einige sind aufgrund von NDAs kondensiert dargestellt.  
Teil 1.



22



23



24



25



26



27



28



29



30



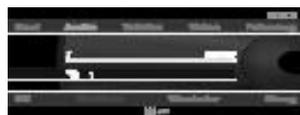
31



32



33



34



35



36



37



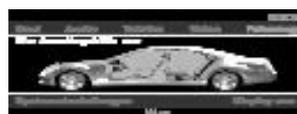
38



39



40



41



42



43

Abbildung B.2.: Testbilder mit 8:3 Seitenverhältnis. Einige sind aufgrund von NDAs kondensiert dargestellt.  
Fortsetzung.

## C. Source-Code Auszüge

```
1 function [led_influence] = decode_lsf(sp_position_x, sp_position_y, current_led_params,
   lsf_model3_led_center, lsf_model7_led_center, LSF_models_SP)
2
3 % Entsprechen der Zuordnung zur "Lage"
4 lsf_model = current_led_params.model;
5 mirror_bit_y_axis = current_led_params.m_y;
6 mirror_bit_x_axis = current_led_params.m_x;
7 border_top = current_led_params.b_top;
8 border_left = current_led_params.b_left;
9
10 sp_resolution_x = 129;
11 sp_resolution_y = 49;
12 % Modelle, Lage
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14 % 1 2 3 3 3 3 3 3 3
15 % 4 5 6 6 6 6 6 6 6
16 % 7 8 9 9 9 9 9 9 9
17 % 7 8 9
18 % 7 8 9
19 % 4 5 6
20 % 1 2 3 3 3 3 3 3 3
21 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
22 if (lsf_model == 1 || lsf_model == 2 || lsf_model == 4 || lsf_model == 5)
23     if(mirror_bit_x_axis)
24         lsf_addr_y = sp_resolution_y - sp_position_y;
25     else
26         lsf_addr_y = sp_position_y;
27     end
28     if (mirror_bit_y_axis)
29         lsf_addr_x = sp_resolution_x - sp_position_x ;
30     else
31         lsf_addr_x = sp_position_x;
32     end
33 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
34 elseif (lsf_model == 3 || lsf_model == 6)
35     if (mirror_bit_x_axis)
36         lsf_addr_y = sp_resolution_y - sp_position_y;
37     else
38         lsf_addr_y = sp_position_y;
39     end
40     if (mirror_bit_y_axis)
41         lsf_addr_x = sp_resolution_x - sp_position_x;
42         led_sp_addr_x = sp_resolution_x - current_led_params.sp_x;
43     else
44         lsf_addr_x = sp_position_x;
45         led_sp_addr_x = current_led_params.sp_x;
46     end
47     if (~border_left) % wenn Bit nicht gesetzt ist, muss eventuell gespiegelt werden
48         if (lsf_addr_x - led_sp_addr_x > 0)
49             lsf_addr_x = lsf_model3_led_center.sp_x + lsf_addr_x - led_sp_addr_x;
50         else
51             lsf_addr_x = lsf_model3_led_center.sp_x + led_sp_addr_x - lsf_addr_x;
52         end
53     end
54 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```

55 elseif (lsf_model == 7 || lsf_model == 8)
56     ...
57     % analog zu lsf_modell == 3 und 6
58     ...
59     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
60 elseif (lsf_model == 9)
61     if (mirror_bit_x_axis)
62         % analog zu oben
63     end
64     if (mirror_bit_y_axis)
65         % analog zu oben
66     end
67     if (~border_left)
68         % analog zu oben
69     end
70     if (~border_top)
71         % analog zu oben
72     end
73 end
74 % x-Position ausserhalb der vollen SP-Modell-Aufloesung
75 pos_x_over_square_size = lsf_addr_x - sp_resolution_y;
76 if (pos_x_over_square_size > 0)
77     lsf_addr_x = ceil(pos_x_over_square_size/2);
78     lsf_addr_y = ceil(lsf_addr_y/2);
79     % In LSF_models_SP{:, 2} sind die reduzierten SPs gespeichert
80     led_influence = LSF_models_SP{lsf_model, 2}((lsf_addr_y-1)*(sp_resolution_x-sp_resolution_y)/2
81         + lsf_addr_x);
82 else
83     % In LSF_models_SP{:, 1} sind die vollen SPs gespeichert
84     led_influence = LSF_models_SP{lsf_model, 1}((lsf_addr_y-1)*sp_resolution_y + lsf_addr_x);
85 end
end

```

Codeauszug C.1: Code-Ausschnitt LSF-Decoders in Matlab. Ohne Pipeline-Optimierung.

```

1 000000000000001
2 000100101100000
3 001001011000001
4 001110000100010
5 010010110000011
6 010111011100100
7 011100001000101
8 100000110100110
9 100101100000111
10 000000000000001
11 000001111100111
12 000011111001111
13 000101110110111
14 000111110011111
15 001001110000111
16 001011101101111
17 001101101010111
18 001111100111111

```

Codeauszug C.2: Inhalt des Offset-ROMs für die Modelladressenberechnung (*LSF\_addressing\_offsets*). Ab Zeile 10 sind die Adressoffsets für den ROM mit reduzierter Auflösung gespeichert.

```

1  module precalc_lsf_addr_1D #(parameter BIT_DEPTH = 8, parameter INDEX_SHIFT = 1)(
2  input clk,
3  input [BIT_DEPTH-1:0] c_led_SP_pos,
4  input [BIT_DEPTH-1:0] c_SP,
5  input [BIT_DEPTH-1:0] model_pos_const,
6  input [BIT_DEPTH-1:0] SP_max,
7  output reg [BIT_DEPTH-1:0] lsf_addr_1,
8  output reg [BIT_DEPTH-1:0] lsf_addr_2,
9  output reg [BIT_DEPTH-1:0] lsf_addr_3,
10 output reg [BIT_DEPTH-1:0] lsf_addr_4,
11 output reg [BIT_DEPTH-1:0] led_sp_addr_mirror,
12 output reg [BIT_DEPTH-1:0] led_sp_addr_NOT_mirror);
13 always @(posedge clk)
14 begin
15 lsf_addr_1 <= SP_max - c_SP + INDEX_SHIFT;
16 lsf_addr_2 <= c_SP;
17 lsf_addr_3 <= model_pos_const + c_SP - c_led_SP_pos;
18 lsf_addr_4 <= model_pos_const + c_led_SP_pos - c_SP;
19
20 led_sp_addr_mirror <= SP_max - c_led_SP_pos + INDEX_SHIFT;
21 led_sp_addr_NOT_mirror <= c_led_SP_pos;
22 end
23 endmodule

```

Codeauszug C.3: Taktweise Berechnung der vorprozessierten Adressen.

```

1  module conditional_case #(parameter BIT_DEPTH = 8)(
2  input clk_i,
3  input mirror_bit_i,
4  input border_bit_i,
5  input [BIT_DEPTH-1:0] lsf_addr_1_i,
6  input [BIT_DEPTH-1:0] lsf_addr_2_i,
7  input [BIT_DEPTH-1:0] lsf_addr_3_i,
8  input [BIT_DEPTH-1:0] lsf_addr_4_i,
9  input [BIT_DEPTH-1:0] lsf_addr_pos_i,
10 input [BIT_DEPTH-1:0] led_sp_addr_i,
11 output reg [BIT_DEPTH-1:0] lsf_addr_o); // entspricht der linearisierten Adresse
12 always @(posedge clk_i)
13 begin
14 if ({mirror_bit_i,border_bit_i,(lsf_addr_pos_i > led_sp_addr_i)} == 3'b000)
15 lsf_addr_o <= lsf_addr_4_i;
16 if ({mirror_bit_i,border_bit_i,(lsf_addr_pos_i > led_sp_addr_i)} == 3'b001)
17 lsf_addr_o <= lsf_addr_3_i;
18 if ({mirror_bit_i,border_bit_i} == 2'b01)
19 lsf_addr_o <= lsf_addr_2_i;
20 if ({mirror_bit_i,border_bit_i,(lsf_addr_pos_i > led_sp_addr_i)} == 3'b100)
21 lsf_addr_o <= lsf_addr_3_i;
22 if ({mirror_bit_i,border_bit_i,(lsf_addr_pos_i > led_sp_addr_i)} == 3'b101)
23 lsf_addr_o <= lsf_addr_4_i;
24 if ({mirror_bit_i,border_bit_i} == 2'b11)
25 lsf_addr_o <= lsf_addr_1_i;
26 end
27 endmodule

```

Codeauszug C.4: Submodul zur fallabhängigen LSF-Adressberechnung.



# Abkürzungsverzeichnis und Akronyme

<b>AM</b>	Amplitudenmodulation
<b>APL</b>	Average Pixel Level
<b>ASIC</b>	Application-Specific Integrated Circuit (Anwendungsspezifische integrierte Schaltung)
<b>ASIL</b>	Automotive Safety Integrity Level
<b>BGA</b>	Ball Grid Array (Kugelgitteranordnung)
<b>BLU</b>	Backlight-Unit
<b>CABC</b>	Content Adaptive Backlight Control
<b>CCD</b>	Charge-Coupled Device (Ladungsgekoppeltes Bauteil)
<b>CID</b>	Center Information Display (Zentralanzeige)
<b>CIE</b>	Commission Internationale de l'Éclairage (Internationale Beleuchtungskommission)
<b>CLB</b>	Configurable Logic Block (konfigurierbarer Logik Block)
<b>CMOS</b>	Complementary metal-oxide-semiconductor
<b>CRT</b>	Cathode Ray Tube (Kathodenstrahlröhrenbildschirm)
<b>CSF</b>	Contrast-Sensitivity Function (Kontrastempfindlichkeitsfunktion)
<b>DBEF</b>	Dual Brightness Enhancement Film von 3M™
<b>DFD</b>	Deutsches Flachdisplay-Forum e.V.
<b>EDID</b>	Extended Display Identification Data
<b>EEPROM</b>	Electrically Erasable Programmable Read-only Memory (elektrisch löschbarer programmierbarer Nur-Lese-Speicher)
<b>EOTF</b>	Electro-Optical Transfer Function
<b>FF</b>	Flip-Flop (bistabile Kippstufe)
<b>FMC</b>	FPGA Mezzanine Card
<b>FPD</b>	Flat Panel Display Link
<b>FPDM</b>	Flat Panel Display Measurements Standard
<b>FPGA</b>	Field Programmable Gate Array (Feld programmierbare Gatter-Anordnung)
<b>FPS</b>	Frames per Second
<b>FWHM</b>	Full Width at Half Maximum (Halbwertsbreite)

<b>GPU</b>	Graphics Processing Unit (Grafikprozessor)
<b>HCFL</b>	Hot Cathode Fluorescent Lamp (Glühkathoden-Fluoreszenzlampe)
<b>HDL</b>	Hardware Description Language (Hardwarebeschreibungssprache)
<b>HDMI</b>	High Definition Multimedia Interface
<b>HDR</b>	High Dynamic Range (hoher Dynamikumfang)
<b>HMI</b>	Human-Machine-Interface (Mensch-Maschine-Schnittstelle)
<b>HUD</b>	Head-up-Display
<b>HVS</b>	Human Visual System (menschliche visuelle System)
<b>I2C</b>	Inter-Integrated Circuit (I-Quadrat-C)
<b>IC</b>	Integrated Circuit (Integrierter Schaltkreis)
<b>ICD</b>	Instrument Cluster Display (Kombi-Instrument Anzeige)
<b>IDMS</b>	Information Display Measurement Standard
<b>IEC</b>	International Electrotechnical Commission (Internationale Elektrotechnische Kommission)
<b>IP</b>	Intellectual Property
<b>IPS</b>	In-Plane-Switching
<b>IQA</b>	Image Quality Assessment (Bildgütebestimmung)
<b>ISO</b>	International Organization for Standardization
<b>JND</b>	Just Noticeable Difference (differentielle Wahrnehmbarkeitsschwelle)
<b>LCD</b>	Liquid Crystal Display (Flüssigkristallanzeige)
<b>LED</b>	Light-Emitting Diode
<b>LME</b>	Lehrstuhl für Mikroelektronik an der Universität des Saarlandes
<b>LSB</b>	Least Significant Bit
<b>LSF</b>	Light-Spread-Function
<b>LTSP</b>	Low Temperature Poly Silicon
<b>LUT</b>	Lookup-Table (Umsetzungstabelle)
<b>LVDS</b>	Low Voltage Differential Signaling
<b>MATLAB</b>	MATrix LABoratory
<b>MOSFET</b>	Metal-Oxide-Semiconductor Field-Effect Transistor
<b>MSB</b>	Most Significant Bit
<b>NDA</b>	Non-Disclosure Agreement (Vertraulichkeitsvereinbarung)
<b>OEM</b>	Original Equipment Manufacturer (Originalausrüstungshersteller)
<b>OETF</b>	Opto-Electronic Transfer Function
<b>OLED</b>	Organic Light-Emitting Diode

<b>PCB</b>	Printed Circuit Board (Platine, gedruckte Schaltung)
<b>Pixel</b>	Kunstwort aus <i>picture</i> und <i>element</i>
<b>PLL</b>	Phase-Locked Loop (Phasenregelschleife)
<b>PWM</b>	Pulse Width Modulation (Pulsweitenmodulation)
<b>RAM</b>	Random Access Memory
<b>RMSE</b>	Root-Mean-Square Error (Wurzel der mittleren Fehlerquadratsumme)
<b>ROM</b>	Read-Only Memory (Festwertspeicher)
<b>RTL</b>	Register Transfer Level (Registertransferebene)
<b>SDR</b>	Standard Dynamic Range
<b>SoA</b>	State-of-the-art (Stand der Technik)
<b>SPI</b>	Serial Peripheral Interface
<b>SPWG</b>	Standard Panel Working Group
<b>SRAM</b>	Static Random Access Memory
<b>SSC</b>	Sorted Sector Covering
<b>TCON</b>	Timing-Controller
<b>TFT</b>	Thin-Film-Transistor (Dünnschichttransistor)
<b>TMDS</b>	Transition-Minimized Differential Signaling
<b>UART</b>	Universal Asynchronous Receiver Transmitter
<b>USP</b>	Unique Selling Point (Alleinstellungsmerkmal)
<b>UUT</b>	Unit Under Test (Prüfling)
<b>VA</b>	Vertical Alignment
<b>VESA</b>	Video Electronics Standards Association



# Formelzeichen und Variablen

$\Delta_v(l)$	Fehlende Differenz des Wertes der LED $l$ um $\Delta_{\check{L}}(i, j)$ zu genügen.
$\Delta_{\check{L}}(i, j)$	Fehlender Lichtbeitrag in Condensierzelle mit dem Wert $c_{i,j}$ an der Stelle $(i, j)$ .
$N_{\text{iter}}$	Anzahl der Iterationen über die Pixelorder.
$f_p$	Frequenz der Pixelclock.
$f_s$	Frequenz der Pixelclock.
$N_C$	Gesamtanzahl der Condensierzellen.
$N_C^c$	Anzahl der Condenser Spalten in horizontaler Richtung.
$N_C^r$	Anzahl der Condenser Zeilen in vertikaler Richtung.
$p_{i,j}^C$	Pixel mit Position $(i, j)$ innerhalb einer Condensierzelle $C$ .
$N_L^c$	Anzahl der LED Spalten in Matrixanordnung.
$N_L^r$	Anzahl der LED Zeilen in Matrixanordnung.
$N_L$	Gesamtanzahl der LEDs.
$\tilde{N}_L$	Anzahl der berücksichtigten LEDs.
$\text{LED}_{p,l}^M$	LED mit dem Index $l$ mit dem höchsten Einfluss im Pixel mit dem Index $p$ , auch Haupt-LED.
$\text{LED}_l$	LED mit dem linearen Index $l$ .
$\text{LED}_{r,c}$	LED an der Position $(r, c)$ bezogen auf LED-Spalten bzw. Zeilen.
$\mathcal{L}$	Menge der LEDs in der BLU.
$\mathcal{M}_{i,j}$	Menge der berücksichtigten LEDs beim Pixel bzw. Condensierzelle $(i, j)$ .
$L_{\Sigma,t}^{\text{abs}}$	All-On in Matrix Darstellung mit LC-Transmission $t$ . Einheit $\frac{\text{cd}}{\text{m}^2}$ .
$\check{L}_p^{\text{max}}$	Maximaler Einfluss in Pixel $p$ .
$\check{L}$	Mittelwert aller Einflüsse.
$\check{L}_{\text{total}}^{\text{max}}$	Höchster vorhandener Einfluss.
$\check{L}^{\text{max}}$	Mittelwert der maximalen Einflüsse pro Pixel.
$\check{L}_{(i,j),l}$	Einfluss der LED mit dem Index $l$ auf den Pixel $(i, j)$ .
$\check{L}^{\text{max}}$	Maximaler Einfluss in einem Pixel in Matrixdarstellung.
$\check{L}_{p,l}$	Einfluss der LED mit dem Index $l$ auf den Pixel mit dem linearen Index $p$ .
$\check{L}$	Einflussmatrix.
$\check{L}_l$	Einfluss der LED mit dem Index $l$ auf alle Pixel.
$L_{(i,j),l}^{\text{abs}}$	Absolute Leuchtdichte der LED mit dem Index $l$ in Pixel $(i,j)$ .

$L_l^{\text{abs}}$	Absolute Leuchtdichte der LED mit dem Index $l$ in Matrix Form.
$L_{p,l}^{\text{abs}}$	Absolute Leuchtdichte der LED mit dem Index $l$ in dem Pixel mit dem linearen Index $p$ .
$\check{\Lambda}$	Beschreibt die <i>Lokalität einer BLU</i> .
$L_{(i,j),l}^{\text{model}}$	Modellierte Leuchtdichte in Pixel $(i,j)$ der LED mit dem Index $l$ .
$L_{(i,j)}^{\text{abs}}$	Absolute Leuchtdichte in Pixel $(i,j)$ .
$\Delta_{\check{L}^{\text{max}}}^{\check{L}^{\text{total}}}$	Differenz zwischen $\check{L}_{\text{total}}^{\text{max}}$ und $\check{L}^{\text{max}}$ .
$r_{50\%}$	Verhältnis zwischen FWHM und LED-Pitch.
$\check{\mathbf{B}}$	Skalarfeld des virtuellen Backlights.
$N'_L$	Anzahl der modellierten LEDs.
$\mathcal{L}'$	Menge der modellierten LEDs.
$\check{L}_l^{\text{SP}'}$	Modellbasierter Einfluss in Sample-Point Auflösung der LED mit dem Index $l$ auf alle Stützstellen der Bildebene.
$\check{L}'_l$	Modellbasierter Einfluss der LED mit dem Index $l$ auf alle Pixel.
$M_i^{\text{SP}'}$	Skalarfeld des LSF Modells mit linearem Index $i$ , basieren auf Stützstellen.
$M'_i$	Skalarfeld des LSF Modells mit linearem Index $i$ .
$M'_i$	LSF Modell mit linearem Index $i$ .
$P_{\text{black}}$	Anteil der schwarzen Pixel im Bild $< L_{\text{th}}$
$N_P$	Gesamtanzahl der Pixel.
$N_P^c$	Anzahl der Pixel Spalten in horizontaler Richtung.
$N_P^r$	Anzahl der Pixel Zeilen in vertikaler Richtung.
$\mathcal{P}$	Menge der Pixel des Panels.
$N_{\text{SP}}$	Gesamtanzahl der Sample-Points.
$N_{\text{SP}}^c$	Anzahl der Sample-Points Spalten in horizontaler Richtung.
$N_{\text{SP}}^r$	Anzahl der Sample-Points Zeilen in vertikaler Richtung.
$p_{i,j}^{\text{SP}}$	Pixel mit Position $(i, j)$ innerhalb eines Sample-Points $SP$ .
$\check{\mathbf{B}}^{\text{SP}}$	Skalarfeld des virtuellen Backlights in SP-Auflösung.
$\mathbb{1}^{n \times m}$	$n \times m$ -Einsmatrix

# Tabellenverzeichnis

2.1. Radiometrische und entsprechende fotometrische Größen in der Übersicht. . . . .	14
4.1. Vergleich der resultierenden $\check{\Lambda}$ angegeben in %. . . . .	85
4.2. Modellparameter für die Generierung der LSFs für konstanten $r_{50\%}$ . . . . .	111
4.3. Maximaler Einfluss der 6 ausgewählten LSF-Formen mit konstantem $r_{50\%}$ und resultierende Leistungseinsparung. . . . .	115
4.4. Modellparameter für die Generierung der LSFs für konstanten maximalen Einfluss. . . . .	118
4.5. Reduktion des Leistungsumsatzes mit Local-Dimming. . . . .	123
4.6. Verschiedene Breiten zu LED-Pitch Verhältnisse der 6 ausgewählten LSF-Formen mit festgelegtem maximalen Einfluss. . . . .	124
4.7. Pearson-Korrelationskoeffizienten der verschiedenen LSF-Breiten zu Pitch Verhältnissen zu Uniformitätsmaß aus Abbildung 4.61. . . . .	124
4.8. Vergleich der aller bisheriger $\check{\Lambda}$ angegeben in %. . . . .	125
4.9. Anzahl <i>schwarzer</i> Pixel $< 1/K \cdot L_{\max}$ des HMI Testbildes 3 bei verschiedener LED Anzahl. . . . .	127
5.1. SRAM Bedarf der SoA Implementierung für den Condenser- und SP-RAM in kbit. . . . .	135
5.2. Median und Standardabweichung der Anzahl an lichtbeitragenden LEDs basierend auf Abbildung 5.9. In 12 Bit Auflösung gespeichert. . . . .	139
5.3. Median und Standardabweichung der Beitragssummen in 12 Bit Auflösung. . . . .	139
5.4. Zuordnung der Modell- zu LED-Nummern. . . . .	151
5.5. Mittleres Powersaving und initiale Performance der Sortierung bei einer Iteration. . . . .	166
5.6. Zuordnung verschiedener Modell-Adressen in Abhängigkeit der LED und SP-Lage für das Modell 9. . . . .	180
6.1. LED-Werte (Duty-Cycle) an den markierten Positionen aus Abbildung 6.21. . . . .	213
6.2. RMSE bei $7 \times 7$ LEDs Betrachtung mit Fehlerkorrekturen der SP-Berechnung basierend auf den beiden Testvideos. . . . .	225
7.1. Zusammenfassende Übersicht über den SRAM Bedarf des Local-Dimming Prozessors. . . . .	232
7.2. Mittleres Powersaving der Testbilder aus Anhang B für verschiedene Parametersätze der 240er BLU. . . . .	233



# Abbildungsverzeichnis

1.1. Beispiele verschiedener Displays im Automobil. . . . .	1
1.2. Entwicklung des Absatzes von ICDs und CIDs. . . . .	2
1.3. Zukünftige Entwicklung im Cockpit. . . . .	2
1.4. Durchschnittliches Alter von Pkws in Deutschland. . . . .	3
1.5. EU Grenzwerte für den CO <sub>2</sub> Ausstoß bei Neuwagen. . . . .	4
1.6. Struktur der Monografie mit Lese- und Erkenntnisflow. . . . .	8
2.1. Elektromagnetisches Spektrum. . . . .	12
2.2. Eldim UMaster und SpectraScan PR-740 im Dunkelraum des LME. . . . .	15
2.3. Aufbau des menschlichen Auges. . . . .	15
2.4. Netzhaut mit Dichteverteilung der Zapfen und Stäbchen. . . . .	16
2.5. Relative spektrale Hellempfindlichkeitsgrad $V(\lambda)$ . . . . .	17
2.6. Gamma Prozessierung im Datenpfad. . . . .	19
2.7. Messergebnisse einer Gamma-Messung. . . . .	19
2.8. Beziehungen und Zusammenhänger der Farbräume. . . . .	20
2.9. Aufbau eines TFTs. . . . .	21
2.10. Visualisierung der nematischen Mesophase des Flüssigkristalls. . . . .	21
2.11. Schematischer Aufbau einer TN-LC-Zelle. . . . .	22
2.12. Schematischer Aufbau eines LCDs mit Hintergrundbeleuchtung. . . . .	23
2.13. Schematische Anordnung der LEDs in einer BLU. . . . .	23
2.14. Optische Distanz bei der Direct-Lit BLU. . . . .	24
2.15. Aktiv-Matrix Ansteuerung des LC-Panels. . . . .	24
2.16. Vereinfachtes Diagramm zur Funktionsweise der Lichtemission bei einem organischen Halbleiter. . . . .	25
2.17. Multilayer-Struktur einer OLED sowie schematische Energieniveaus. . . . .	25
2.18. OLED Aktiv-Matrix Ansteuerung mit der 2T1C Realisierung. . . . .	26
2.19. Störendes Light-Leakage eines CIDs. . . . .	27
2.20. Messung des Light-Leakage mit der Eldim UMaster und resultierender lokaler Kontrast. . . . .	27
2.21. Gamma-Messung und entsprechende Farbörter eines vermessenen LC-Panels. . . . .	28
2.22. Grundprinzip eines Dimming-Systems. . . . .	29
2.23. Integration eines Dimming Algorithmus zwischen der EOTF und OETF. . . . .	30
2.24. Überblick der verschiedenen Dimming-Arten und ihren Beziehungen. . . . .	31
2.25. Beispiel der verschiedenen Dimming-Methoden anhand der Saarschleife. . . . .	32
2.26. Verschiedene Darstellungsmöglichkeiten von LSFs. . . . .	33
2.27. Beispiel der Pixelkompensation anhand der <i>Saarschleife</i> . . . . .	35
2.28. Aufbau und Zusammensetzung eines LCD-Moduls. . . . .	37
2.29. Pixelpipeline mit Beispielwerten sowie Aufteilung in zwei parallele Pipes. . . . .	38
2.30. Timing Diagramm eines Bilddatenstromes. . . . .	39

2.31. Grundlegende Struktur und Komponenten eines FPGAs. . . . .	40
3.1. Vergleich eines HMI mit einem natürlichen Bild und zugehörige Histogramme. . . . .	42
3.2. Kombinationen der Symbol-/Hintergrundfarbe nach DIN EN ISO 15008:2017-07. . . . .	43
3.3. <i>Magic Circle</i> der Zusammenhänge der Optimierung des Displayeigenschaften. . . . .	43
3.4. Deutsches Kooperationsmodell mit DFF. . . . .	44
3.5. Ein 12,3 Zoll automotive OLED-Display von <i>BOE</i> . . . . .	45
3.6. Aufnahme eines <i>eingebraunten</i> OLED Displays. . . . .	46
3.7. Benchmark zwischen einem OLED-TV und einem LC-TV mit Local-Dimming. . . . .	46
3.8. Veranschaulichung einer histogrammbasierten Global-Dimming Methode. . . . .	47
3.9. Manipulation der Gammakurve zum Zweck des Global-Dimmings. . . . .	48
3.10. Aufnahme (Fotokamera) eines HMI auf dem Prototyp. . . . .	49
3.11. Ausschnitt eines HMI Testbildes ohne und mit Global-Dimming. . . . .	50
3.12. Manipulation Eingangswerte basierend auf wahrnehmungslinearen Daten. . . . .	51
3.13. Blockdiagramm des vorgestellten Global-Dimming Algorithmus. . . . .	52
3.14. Histogramm eines Testbildes zur Verdeutlichung des histogrammbasierten Global-Dimmings. . . . .	53
3.15. LUT für eine bildabhängige Zuordnung von dem <i>Criticality Degree BD</i> auf <i>PQ</i> . . . . .	55
3.16. Zeitlicher Ablauf des Global-Dimming Algorithmus in Hardware. . . . .	56
3.17. Anschluss der Prototypen an das FPGA Board. Das FPGA selbst ist der große IC in der Mitte des PCBs. In diesem Bild ist das zweite Display nicht angeschlossen. . . . .	57
3.18. Prototypenaufbau aus zwei Displays zum direkten Vergleich. . . . .	57
3.19. Powersaving mit variiertem <i>WD</i> bei festen und automatischem <i>PQ</i> basierend auf dem IEC-Testvideo. . . . .	58
3.20. SoA im Vergleich zum hier vorgestellten Global-Dimming Algorithmus. . . . .	58
3.21. Vergleich der mittleren Leuchtdichten für gedimmte und ungedimmte Testbilder und Leistungsumsatz. . . . .	59
3.22. ELDIM Leuchtdichte-Messung. Oben ungedimmt, unten gedimmt mit einem Powersaving von 9,4%. . . . .	59
3.23. Schema des SSC Algorithmus nach SoA. . . . .	62
3.24. Datenflussdiagramm des SSC Algorithmus nach SoA. . . . .	62
3.25. Prinzipieller zeitlicher Ablauf des SSC Algorithmus nach SoA. . . . .	63
3.26. Funktionsweise des Condensers mit Modi-Wahl. . . . .	64
3.27. Visualisierung des Condenserinhaltes für das <i>Saarschleifen</i> Bild für zwei Condensermodi, basierend auf den maximalen Subpixeln. . . . .	66
3.28. Vergleich der Häufigkeiten der Betrachtung einzelner LEDs im Ablauf des SSC-Optimierers einer Edge-Lit BLU. . . . .	68
3.29. SoA Ansatz durch <i>Downsampling</i> des Bildinhalts. . . . .	69
3.30. Algorithmus für die LED-Wert-Berechnung nach von Seetzen u. a. . . . .	70
3.31. Vergleich der Prozessierungsrichtung der klassischen Local-Dimming Ansätzen mit dem SSC. . . . .	71
3.32. Schema und Visualisierung der Processing Prozedur für Direct-Lit. . . . .	71
3.33. Schematische Darstellung der Direct-Drive-Ansteuerung von 16 LEDs. . . . .	72
3.34. Vergleich der zeitlichen LED-Ansteuerung im Multiplexverfahren. . . . .	73
3.35. Schematische Darstellung der Direct-Drive Methode unter Nutzung von Zeilenmultiplex. . . . .	73

4.1. BLU Prototyp mit 319 LEDs. . . . .	76
4.2. Abhängigkeiten und Beziehungen der BLU Design-Parameter . . . . .	76
4.3. LED Nummerierung und Anordnung. . . . .	78
4.4. Edge-Lit mit 7 LEDs an der Unterkante. . . . .	79
4.5. Direct-Lit mit 133 LEDs ( $7 \times 19$ ). . . . .	79
4.6. Direct-Lit mit 319 LEDs ( $11 \times 29$ ). . . . .	80
4.7. Resultierende Einflüsse zu Abbildungen 4.4 bis 4.6. . . . .	81
4.8. Querschnitte durch die Einflüsse. . . . .	81
4.9. Visualisierung der Haupt-LED Zuordnung beim Edge-Lit Gerät mit 7 LEDs und Direct-Lit mit 133 LEDs. . . . .	82
4.10. Häufigkeit der Haupt-LED Zuordnung beim 133er Direct-Lit. . . . .	82
4.11. Maximale Einflüsse der Haupt-LEDs pro Pixel beim Edge-Lit Gerät mit 7 LEDs und Direct-Lit mit 133 LEDs. . . . .	83
4.12. Verteilung der maximalen Einflüsse mit einer Bin-Weite von 0,5%. . . . .	83
4.13. Lokalitäten mit steigender LED Anzahl und verschiedene $\Delta \frac{\tilde{I}_{total}^{max}}{\tilde{I}_{max}}$ . . . . .	85
4.14. Backlight des Hisense 65XT910 mit $12 \times 20$ LEDs. . . . .	86
4.15. Hisense 240er Backlight mit 3 aktivierten Dimming-Zonen. . . . .	87
4.16. Simulierte LSF für eine 8:3 BLU mit 216 LEDs. Dargestellt sind die Simulationsdaten einer einzelnen LED. . . . .	87
4.17. Zentrale LED einer simulierten 216er BLU. Die LED-Positionen sind mit roten Punkten markiert. . . . .	88
4.18. Messungen des All-Ons sowie der zentralen Dimming Zone, gemessen vom Projektpartner. Dargestellt sind auch die markierten Querschnitte. . . . .	88
4.19. Foto einer BLU (ohne LC-Panel) mit Schachbrettmuster ( $10 \times 24$ Zonen). . . . .	89
4.20. Übersicht und Querschnitte durch LSF und resultierende Einflüsse der obigen 240er BLU. . . . .	89
4.21. Vergleich des Einflusses der beiden vorgestellten 240er BLUs mit Markierung der LED Zentren in horizontaler Richtung. . . . .	90
4.22. Vergleich der vorgestellten 133er, 319er und 240 BLUs. . . . .	91
4.23. Testbild und resultierende Leuchtdichte basierend auf einer quadratischen LSF. . . . .	91
4.24. Visualisierung der Entstehung von sichtbaren Blöcken durch zu steile LSFs. . . . .	93
4.25. Querschnitt durch das absolute All-On der 133 BLU mit Fit einer sinusförmigen Funktion. . . . .	94
4.26. Vergleich der vorgeschlagenen Messpunkte auf einem zu vermessenden Display. . . . .	96
4.27. Messpunkte zur Berechnung der lokalen Uniformität. . . . .	97
4.28. Verbreiter Ablauf des BLU-Designs. . . . .	99
4.29. Vorgeschlagener Ablauf des BLU-Designs mit Berücksichtigung von Local-Dimming. . . . .	100
4.30. Vergleich einer LSF mit und ohne Diffusor. . . . .	101
4.31. Beispiel LSF einer BLU mit 240 Kavitäten und ein einfaches Modell realisiert durch eine Gauß-Verteilung. . . . .	102
4.32. Exemplarischer LSF-Querschnitt einer Messung. . . . .	103
4.33. Kurvenschar von $M_1(x)$ . . . . .	104
4.34. Vergleich der Bestandteile des eingeführten Modells. . . . .	104
4.35. Vergleich Modell mit realer LSF. . . . .	105
4.36. FWHM zu LED-Pitch Verhältnis. . . . .	106

4.37. Vergleich zweier realer BLUs mit verschiedenem LED-Pitch und identischer Halbwertsbreiten der LSFs. . . . .	106
4.38. Vergleich zweier simulierter BLUs mit verschiedenem LED-Pitch und identischer Halbwertsbreiten der LSFs. . . . .	107
4.39. Maximaler Einfluss in Abhängigkeit vom FWHM/LED-Pitch Verhältnis. . . . .	107
4.40. Maximale und mittlere Einflüsse in Abhängigkeit von der Anzahl der LEDs. . . . .	108
4.41. Mittlerer Einfluss und All-On bei fester Kavitätenanzahl und variiertem $r_{50\%}$ . . . . .	109
4.42. Ablauf der LSF-Modell Generierung. . . . .	110
4.43. Variation der Modellparameter und resultierender Einfluss. . . . .	112
4.44. Normierte Leuchtdichte simuliert aus 6 verschiedenen Modellen. . . . .	112
4.45. All-On von $L_6$ mit LED-Positionen. . . . .	113
4.46. Querschnitt durch 2 LSFs des Modells $L_6$ . . . . .	113
4.47. Querschnitte durch obige All-Ons der LSF Modelle. . . . .	114
4.48. $L_1$ und $L_3$ unter Annahme eines zufälligen Fehlers bei der LED-Platzierung. . . . .	115
4.49. Genutzte Messpunkte zur Bestimmung der globalen Uniformität. . . . .	116
4.50. Nicht-Uniformitäten bei steigendem LED-Jitter. . . . .	116
4.51. Local-Dimming: resultierendes Backlight zum Testbild Nummer 3. . . . .	117
4.52. Variation der Parameter A, B und C für eine 216er BLU bei konstantem maximalen Einfluss. . . . .	118
4.53. Resultierender Einfluss von $M_1 \dots M_6$ berechnet nach Formel (2.15) (Seite 33). . . . .	119
4.54. Normierte Leuchtdichte simuliert aus 6 verschiedenen Modellen. . . . .	120
4.55. Visualisierung der beiden Messpunkte zur Bestimmung von $U_2$ . . . . .	120
4.56. Querschnitt durch die rote Linie aus Abbildung 4.55. . . . .	121
4.57. Auswirkung auf die Uniformität bei nicht akkurater LED-Platzierung. . . . .	121
4.58. Relative Abweichung als vertikaler Querschnitt durch die Mitte der BLU. . . . .	122
4.59. $r_{50\%}$ gegen $\bar{D}$ (rot) und $\bar{G}$ (blau). . . . .	122
4.60. Local-Dimming: resultierendes Backlight zum Testbild Nummer 3. . . . .	123
4.61. Diverse LSF-Breiten zu LED-Pitch Verhältnisse vs. Nicht-Uniformität $1 - U_2$ . . . . .	125
4.62. Resultierendes Powersaving von der Quadratwurzel der LED-Anzahl . . . . .	126
4.63. Beispiele simulierter Backlights mit unterschiedlicher LED-Anzahl. Jeweils 96, 319 und 2400 LEDs. . . . .	126
5.1. Verschiedene Condensergrößen im Vergleich. . . . .	130
5.2. APL in Abhängigkeit von der Quadratwurzel der Condenserzellenanzahl. . . . .	130
5.3. Beispiel einer Sample-Point basierenden LSF Speicherung einer Direct-Lit LSF sowie die resultierende bilineare Interpolation. . . . .	131
5.4. RMSE resultierend aus Sample-Point basierenden LSF Speicherung mit anschließender bilinearen Interpolation. . . . .	132
5.5. SSC Berechnungszeit vs. Quadratwurzel der LED-Anzahl für verschiedene Condensergrößen. . . . .	133
5.6. Pixelorder ROM (als SRAM) Bedarf vs. Quadratwurzel der LED-Anzahl für verschiedene Condensergrößen. . . . .	134
5.7. Markierte LEDs aus $\mathcal{M}_{i,j}$ mit $ \mathcal{M}_{i,j}  = 25$ für die rote Condenserzelle. . . . .	137
5.8. Anzahl der LEDs mit Lichtbeitrag pro Pixel für verschiedene $L_{th}^{min}$ . . . . .	138
5.9. Verteilungen der einflussnehmenden LEDs für verschiedene $L_{th}^{min}$ sowie die entsprechende Verteilung der Beitragssummen. . . . .	138

5.10. Reduktion des Leistungsumsatzes für verschiedene $\tilde{N}_L$ und bei verschieden ausgelegten Iterationszahlen $N_{iter}$ der Pixelorder. . . . .	139
5.11. Reduktion des Leistungsumsatzes bei konstantem maximalen Einfluss von 30 % für verschiedene $r_{50\%}$ und $\tilde{N}_L$ . . . . .	140
5.12. Lage der LEDs der 133 BLU. . . . .	141
5.13. Querschnitte durch die LSFs (4 Zeilen). . . . .	142
5.14. Verdeutlichung des Problems beim Vergleich zwischen LSFs an verschiedenen Positionen. . . . .	143
5.15. Messdaten und resultierender Fit einer Gauss-Superposition ( $R^2 = 0,9998$ ). . . . .	143
5.16. Resultierender Fehler zum Fit aus Abbildung 5.15 (RMSE). . . . .	144
5.17. Auswahl der Modelle und ihre Lage. . . . .	145
5.18. Visualisierung des Problems der fehlenden Werte bei Verschiebung eines Modells und Lösung. . . . .	145
5.19. Beispiel (ein-dimensional) zur Bestimmung der virtuellen Position bezogen auf ein Modell in Abhängigkeit der Pixellage . . . . .	146
5.20. Visualisierung der Mapping-Funktionen am Beispiel der 240er BLU. . . . .	147
5.21. Ablauf zur Erzeugung konsistenter Modelle. . . . .	149
5.22. Beispiel der Bestimmung von LED-Zentren mit 3 beispielhaft ausgewählten LSFs. . . . .	150
5.23. Vergleich des synthetischen All-Ons aus den 9 Messungen basierend auf einer 133 BLU. . . . .	151
5.24. Skalarfelder $M'_i$ der Modelle 1...9 resultierend aus der Normierung mit synthetischem All-On (Panelauflösung). . . . .	152
5.25. Modellbasierende Summe der Einflüsse in jedem Pixel. . . . .	152
5.26. Aufbau und Lage, sowie Relation der Condenserezellen sowie der Sample-Points. . . . .	154
5.27. Blockdiagramm der beiden Methoden zur Erzeugung der Sample-Points der Modelle. . . . .	155
5.28. Stützstellen-basierendes Einflussmodell $M_9$ . . . . .	156
5.29. Visualisierung der Modellierung mit Reduktion der Stützstellendichte für entfernt liegende SPs. . . . .	157
5.30. Abhängigkeitsdiagramm und Implikation auf die Pixelorder. . . . .	159
5.31. Beispiel der Zuordnung der LEDs zu einer Condenserezelle. . . . .	161
5.32. Abarbeitungsreihenfolge der Condenserezellen für ein Edge-Lit Gerät. . . . .	162
5.33. Zuordnung der Haupt-LEDs zu den Condenserezellen. . . . .	162
5.34. Verschiedene Abarbeitungsreihenfolgen der Pixelorder. . . . .	163
5.35. Verteilungen der Haupt-LEDs in den ersten $N_L$ Zeilen der Pixelorder (LED-Nummern linear indiziert). . . . .	164
5.36. Ausgewählte Testbilder (links) und resultierender Condenser-RAM-Inhalt (rechts) dargestellt in 10 Bit Auflösung. . . . .	165
5.37. Potential der Reduktion des Leistungsumsatzes der 6 ausgewählten Sortierungen zu den oben gezeigten Testbildern. . . . .	166
5.38. Beispiel eines möglichen Clusterings der 133er BLU. . . . .	168
5.39. Anwendung der Clusteringmethode auf einer 96-LED BLU. . . . .	169
5.40. Datenflussdarstellung des zu implementierenden Algorithmus. . . . .	170
5.41. Querschnitte der verwendeten BLUs. . . . .	170
5.42. Prototypen mit FPGA-Board und LC-Panel auf 240er BLUs. . . . .	172
5.43. Schematischer Aufbau und Konzeption des Prototyps. . . . .	173
5.44. Verschaltung der TLC-Treiber als <i>Daisy-Chain</i> . . . . .	174
5.45. Abstrahierte Modulübersicht und Hierarchie der IP für die FPGA-Implementierung . . . . .	176
5.46. <i>LSF-Decoder</i> Modul mit Pipelining in RTL Darstellung. . . . .	179

5.47. Linke Hälfte des <i>calc_LSF_address</i> Moduls. Dargestellt sind die ersten beiden Registerstufen.	181
5.48. Fortsetzung des Moduls mit Mittelteil und Ende. Das genutzte Schieberegister (als FIFO) wird genutzt, um die Daten mit dem Ausgang des <i>Multiplizierers</i> synchron zu halten ( $t_5 \mapsto t_{11}$ ).	182
5.49. Visualisierung der Pipeline des <i>LSF-Decoders</i> mit 17 Takten. . . . .	183
5.50. Konzepte der Modulvalidierung des <i>LSF-Decoders</i> . . . . .	185
5.51. (Abstrahierter) Überblick über das Modul <i>Optimizer_Top</i> in RTL-Darstellung. . . . .	186
5.52. Maximale Einflusssumme pro Condensierzelle $(i, j)$ bei Berücksichtigung der $\tilde{N}_L = 25$ LEDs in direkter Nachbarschaft. . . . .	187
5.53. Timing und zeitlicher Ablauf der FPGA-Implementierung. . . . .	190
5.54. Prototyp mit 133er BLU im Gehäuse (3D-Druck von <i>Continental</i> ). Daneben Anzeige des resultierenden Strombedarfs der LEDs mit Hilfe einer GUI. . . . .	190
5.55. Fotos des Prototyps mit 319 LEDs. . . . .	191
5.56. SRAM-Bedarf der SoA Pixelorder im Vergleich mit der neuen Pixelorder sowie der neuen Modellmethode. . . . .	192
5.57. Eldim UMaster Messung eines Testbildes mit resultierendem Backlight. . . . .	192
6.1. <i>Postkarteneffekt</i> auf Local-Dimming Prototyp. . . . .	193
6.2. Eldim UMaster Aufnahme des <i>Black-Mura</i> eines Displays. . . . .	194
6.3. Boosting in Abhängigkeit des gleitenden Mittelwertes Stessen und G. R. van Mourik. . . . .	195
6.4. Berechnung der Boostingparameter und Funktion als Blockdiagramm. . . . .	197
6.5. Parameterauswirkungen auf den tatsächlich möglichen Boost $RB$ . . . . .	198
6.6. Beispiel Local-Dimming ohne Boosting vs. mit Boosting Faktor $RB = 2$ . . . . .	198
6.7. Simulation zweier Testbilder mit einer 240er BLU mit $RB = 2$ vs. ohne Boosting. . . . .	199
6.8. Leuchtdichtemessung (Eldim UMaster) eines Testbildes mit und ohne Boosting sowie Differenzbildung zum Vergleich. . . . .	200
6.9. Beispiel des Halo-Artefakts bei einem Direct-Lit Gerät. . . . .	201
6.10. Visualisierung des Halo Artefakts. . . . .	202
6.11. Vergleich des Kontrastes zwischen einem LCD mit Mini-LED BLU und einem OLED-Display.	203
6.12. Aufteilung in Zonen zur gewichteten Berechnung eines LED-Wertes. . . . .	204
6.13. Fotografie eines HMI Bildes auf einem Edge-Lit LCD mit Überbelichtung. . . . .	205
6.14. Funktionsvisualisierung der LED-Faktoren bzw. des räumlichen Filters. . . . .	206
6.15. An das iterative Design angepasstes Constraint. Grün stellt die aktuelle betrachtete LED dar, welche möglicherweise in diesem Schritt erhöht wurde. Orange die betrachteten Nachbarn.	206
6.16. RTL-Darstellung des <i>factor_assurance</i> Moduls. . . . .	207
6.17. Beispiel zur Auswirkung der erlaubten Abweichung zwischen LED-Nachbarn. . . . .	208
6.18. Beispiel zur Auswirkung des minimalen LED-Wertes auf das Backlight und LED-Lösung. . . . .	210
6.19. Querschnitt durch resultierendes Backlight sowie Vergleich der Lösungen der ersten LED-Zeile zu zum Testbild aus Abbildung 6.13. . . . .	210
6.20. Resultierende Abarbeitungsreihenfolgen der Condensierzellen und die daraus implizierte Reihenfolge der LEDs, welche in den ersten $N_L$ Pixelordereinträgen als Haupt-LEDs betrachtet werden. . . . .	212
6.21. Resultierende LED-Werte im Vergleich zwischen SoA und der hier vorgestellten Methode zur Verbesserung des Schwarzwertes in den Displayecken. . . . .	213

6.22. Realisierungsmöglichkeit der Funktion $r(d, APL)$ zur Anwendung auf die linke obere Ecke des Condensers. . . . .	215
6.23. Vergleich des Condenserinhalts mit und ohne Manipulation zu einem HMI-Testbild. . . . .	215
6.24. Resultierende LED-Werte (einer 240 BLU). . . . .	215
6.25. Leuchtdichtemessungen der Schwarzwertverbesserung in den Displayecken durch Parameteranpassung und Corner-Dimming. . . . .	216
6.26. Leuchtdichteaufnahmen einer 133er BLU im Ursprungszustand (links) und mit Ausfall zweier LEDs (rechts). . . . .	217
6.27. LED-Ausfall einer LED mit und ohne Gegenmaßnahme. . . . .	218
6.28. Berechnungszeiten in Abhängigkeit der Quadratwurzel der LED-Anzahl. . . . .	219
6.29. Abhängigkeitsdiagramm der Berechnungszeit $t_{total}$ . . . . .	220
6.30. Logarithmische Darstellung der Berechnungszeiten $t_{LEDs}$ sowie $t_{SPs}$ . . . . .	221
6.31. Visualisierung des resultierenden $\check{B}^{SP}$ bei Betrachtung einer beschränkten LED-Menge für eine 133er und eine 240er BLU. . . . .	222
6.32. Fehler des virtuellen Backlights bei Reduktion der betrachteten LEDs pro Sample-Point. . . . .	223
6.33. Resultierender RMSE des berechneten Backlights durch Betrachtung einer definierten LED-Zahl für das Testvideo 2. . . . .	224
6.34. Resultierender RMSE des berechneten Backlights durch Betrachtung von $7 \times 7$ LEDs mit Offset-Korrektur pro Sample-Point bei Testvideo 2. . . . .	225
6.35. Resultierende $\check{B}^{SP}$ des Frames 450 von Testvideos 2 mit und ohne Korrekturen der Sample-Points. . . . .	226
6.36. Berechnungszeiten bei 1080 LEDs mit Optimierungsmethoden mit einer Systemclock mit $f_s = 200$ MHz. . . . .	227
6.37. Simuliertes Powersaving mit Visualisierung des Backlights für ein HMI und steigenden BLU-Auflösungen. . . . .	228
7.1. Übersicht der Hardware-Module des Local-Dimming Systems. . . . .	229
7.2. Leuchtdichteaufnahme eines Prototypen. . . . .	230
7.3. Leuchtdichteaufnahmen eines Testbildes im Vergleich zwischen einem SoA Local-Dimming Algorithmus und der neuen IP. . . . .	231
7.4. Visualisierung des Condenserinhaltes und resultierender LED-Werte (8 Bit) und zugehöriges Backlight. . . . .	232
7.5. Einfluss der LSF Lokalität. . . . .	233
7.6. Einflussfaktoren auf die Local-Dimming Effizienz und Performance. . . . .	234
7.7. Display mit einer Auflösung von $1080 \times 7680$ Pixel und einer BLU mit 5096 Dimming Zonen. . . . .	235
7.8. Vergleich zwischen AM und PM Ansteuerung der BLU. . . . .	236
7.9. Funktionsprinzip von Pixeldimming mit Dual-Stack LCD und mögliches Moiré Artefakt. . . . .	237
8.1. Shy Tech Display von <i>Continental</i> mit aktivem Local-Dimming mit Holzdekor. . . . .	241
A.1. Mapping und Schema des FPD-Link mit 2 Kanälen. . . . .	243
A.2. Ablauf einer Design-Implementierung für ein FPGA. . . . .	243
A.3. 9 Messungen der LSFs einer 240 BLU. . . . .	244
A.4. $r_{50\%}$ der 133er BLU. . . . .	244
A.5. $r_{50\%}$ der 319er BLU. . . . .	245

---

A.6. $r_{50\%}$ der 240er BLU mit 240 bzw. 60 Zonen. . . . .	245
A.7. Steigung der 6 ausgewählten Modelle mit konstantem $r_{50\%}$ . . . . .	245
A.8. Final berechnete LED Werte bei verschiedenen Iterationszahlen und Leistungseinsparung . . . . .	246
A.9. Resultierender RMSE des berechneten Backlights durch Betrachtung $7 \times 7$ LEDs mit Offset-Korrektur pro Sample-Point bei Testvideo 1. . . . .	246
B.1. Testbilder mit 8:3 Seitenverhältnis. Teil 1. . . . .	247
B.2. Testbilder mit 8:3 Seitenverhältnis. Fortsetzung. . . . .	248

# Literatur

- [3M 08] 3M Optical Systems Division. *Vikuiti™ Dual Brightness Enhancement Film (DBEF)*. Online. (Zugegriffen am 06.11.2019). 2008. URL: [http://www.opticalfilters.co.uk/includes/downloads/3m/DBEF\\_E\\_DS\\_7516882.pdf](http://www.opticalfilters.co.uk/includes/downloads/3m/DBEF_E_DS_7516882.pdf).
- [Alb10] Marc Albrecht. "Sorted Sector Covering mit Bildkondensierung - Eine universelle Methode zur effizienten Berechnung von Local Dimming LED Backlight". Diss. Saarland University, 2010. DOI: 10.22028/D291-22868.
- [Alb+10] Marc Albrecht, Andreas Karrenbauer, Tobias Jung und Chihao Xu. "Sorted Sector Covering Combined with Image Condensation - An Efficient Method for Local Dimming of Direct-Lit and Edge-Lit LCDs". In: *IEICE Transactions 93-C* (2010), S. 1556–1563. DOI: 10.1587/transele.E93.C.1556.
- [AKX08] Marc Albrecht, Andreas Karrenbauer und Chihao Xu. "A clipper-free algorithm for efficient HW-Implementation of local dimming LED-Backlight". In: *International display research conference (IDRC 08) ; University of Central Florida Orlando, Florida, USA*. 3. Nov. 2008.
- [AKX09] Marc Albrecht, Andreas Karrenbauer und Chihao Xu. "50.3: A Video-Capable Algorithm for Local Dimming RGB Backlight". In: *SID Symposium Digest of Technical Papers 40.1* (2009). Hier wird noch auf die SoA Approches für Local-Dimming verwiesen, S. 753–756. ISSN: 2168-0159. DOI: 10.1889/1.3256897.
- [AX09] Marc Albrecht und Chihao Xu. "Sorted Sector Covering Algorithm with Condensed Image Data and Soft Clipping Extension for Low-Cost Local Dimming Processor and High Power Saving". In: *Proceedings of the International Display Workshops (IDW)* (2009), S. 587–590.
- [All+04] Richard C. Allen, Lockwood W. Carlson, Andrew J. Ouder Kirk, Michael F. Weber, Arthur L. Kotz, Timothy J. Nevitt, Carl A. Stover und Biswaroop Majumdar. "Brightness enhancement film". US-Pat. US6760157B1 (Saint Paul, Minn). Juli 2004. URL: <https://patentimages.storage.googleapis.com/69/8e/c3/a032f58c48de28/US6760157.pdf>.
- [Ana08] Munisamy Anandan. "Progress of LED backlights for LCDs". In: *Journal of the Society for Information Display 16.2* (2008), S. 287–310. DOI: <https://doi.org/10.1889/1.2841864>. URL: <https://sid.onlinelibrary.wiley.com/doi/abs/10.1889/1.2841864>.
- [Aok06] Norio Aoki. "16.3: Invited Paper: Advanced IPS Technology for Mobile Applications". In: *SID Symposium Digest of Technical Papers 37.1* (2006), S. 1087. DOI: 10.1889/1.2433161.
- [Aud] Audi. *Virtuelle Außenspiegel (Audi e-tron)*. Online. (Zugegriffen am 12.09.2021). URL: <https://www.oled-info.com/mercedes-unveiled-its-2022-eqe-350-ev-internal-oled-displays>.
- [Auf] Aufbau des Auges. Online. (Zugegriffen am 26.11.2021). URL: <https://www.lasikon.de/auge/aufbau-des-auges/>.

- [Bai11] Donald G. Bailey. *Design for Embedded Image Processing on FPGAs*. John Wiley & Sons, Singapore, 2011. ISBN: 978-0-470-82849-6.
- [Bal+09] Anders Ballestad, Thomas Wan, Hiroe Li und Helge Seetzen. "Metrics for Local-Dimming Artifacts in High-Dynamic-Range LCDs". In: *Information Display* 25.1 (Jan. 2009), S. 18–22. DOI: 10.1002/j.2637-496x.2009.tb00010.x.
- [Bar16] Norman Bardsley. "Eye Resolution Display". In: Janglin Chen, Wayne Cranton und Mark Fihn. *Handbook of Visual Display Technology*. 2. Aufl. Springer International Publishing, 2016. ISBN: 978-3-319-14345-3.
- [Bar99] Peter G. Barten. *Contrast Sensitivity of the Human Eye and Its Effects on Image Quality*. SPIE, Dez. 1999. DOI: 10.1117/3.353254.
- [Bau+10] J. Bauer, M. Kreuzer, T. Ganz, B. Straub, H. Enigk und N. Boemak. "New ways from visual perception and human contrast sensation to better readable displays in automotive environments". In: *Electronic Displays Conference* (2010).
- [Bau13] Jan Bauer. "EFFIZIENTE UND OPTIMIERTE DARSTELLUNGEN VON INFORMATIONEN AUF GRAFIKANZEIGEN IM FAHRZEUG Situationsadaptive Bildaufbereitungsalgorithmen und intelligente Backlightkonzepte". ISBN 978-3-86644-961-9. Diss. Karlsruher Institut für Technologie, 2013. DOI: 10.5445/KSP/1000031973.
- [BM10] Szymon Beczkowski und Stig Munk-Nielsen. "Led spectral and power characteristics under hybrid PWM/AM dimming strategy". In: *2010 IEEE Energy Conversion Congress and Exposition*. IEEE, Sep. 2010. DOI: 10.1109/ecce.2010.5617930.
- [Ben19] Gennady Benderman. "Entwicklung und Bewertung der E/E-Architektur für hochauflösende Scheinwerfersysteme". Diss. Karlsruher Institut für Technologie (KIT), 2019. 205 S. ISBN: 978-3-7369-7132-5.
- [Ben+09] Jacob Benesty, Jingdong Chen, Yiteng Huang und Israel Cohen. "Pearson Correlation Coefficient". In: *Noise Reduction in Speech Processing*. Springer Berlin Heidelberg, 2009, S. 1–4. DOI: 10.1007/978-3-642-00296-0\_5.
- [Ber91] T. J. T. P. Van Den Berg. "On the relation between glare and straylight". In: *Documenta Ophthalmologica* 78.3-4 (Sep. 1991), S. 177–181. DOI: 10.1007/bf00165678.
- [Ber17] Antonio Bergua. *Das menschliche Auge in Zahlen*. Springer-Verlag GmbH, 29. Juni 2017. ISBN: 978-3-662-47283-5.
- [Bho12] Achintya K. Bhowmik. "Mobile display subsystem power-performance optimization technologies". In: *IMID* (2012), S. 451–452.
- [BHT18] BHTC. *Audi A6 / Center Information Display (CID) - Die Revolution im Innenraum*. Online. (Zugegriffen am 19.01.2022). Behr-Hella Thermocontrol GmbH, 2018. URL: <https://www.bhtc.com/de/news/audi-a6-center-information-display-cid>.
- [Bla16a] Karlheinz Blankenbach. "Luminance, Contrast Ratio, and Gray Scale". In: *Handbook of Visual Display Technology*. Springer International Publishing, 2016, S. 3089–3109. DOI: 10.1007/978-3-319-14346-0\_143.
- [Bla16b] Karlheinz Blankenbach. "Spatial Effects". In: *Handbook of Visual Display Technology*. Springer International Publishing, 2016, S. 3137–3151. DOI: 10.1007/978-3-319-14346-0\_145.

- [Bla18] Karlheinz Blankenbach. "Advanced automotive display measurements: Selected challenges and solutions". In: *Journal of the Society for Information Display* 26.9 (März 2018), S. 517–525. DOI: 10.1002/jsid.625.
- [Bla20] Karlheinz Blankenbach. "Automotive Display Technologies Shine at Display Week 2020". In: *Information Display* 36.5 (Sep. 2020), S. 49–53. DOI: 10.1002/msid.1154.
- [Bla21] Karlheinz Blankenbach. "Automotive Displays: The Race is on Between LCD and OLED!" In: *Information Display* 37.5 (Sep. 2021), S. 19–23. DOI: 10.1002/msid.1244.
- [BHJ16] Karlheinz Blankenbach, Andreas Hudak und Michael Jentsch. "Direct Drive, Multiplex and Passive Matrix". In: *Handbook of Visual Display Technology*. Springer International Publishing, 2016, S. 418–437. DOI: 10.1007/978-3-319-14346-0\_143.
- [Bla+16] Karlheinz Blankenbach, Udo Krüger, Hans-Ulrich Lauer und Martin Zobl. "29-1:Invited Paper: Recent Standardization Efforts and Measurement Procedures of German Automotive OEM and German Flat Panel Forum (DFF)". In: *SID Symposium Digest of Technical Papers* 47.1 (Mai 2016), S. 361–364. DOI: 10.1002/sdtp.10680.
- [BS19] Nikolay Brayanov und Anna Stoyanova. "Review of hardware-in-the-loop -a hundred years progress in the pseudo-real testing". In: *Electrotechnica & Electronica E+E* (1. Dez. 2019).
- [Bro+08] I. N. Bronstein, K. A. Semendjajew, G. Musiol und H. Mühlig. *Taschenbuch der Mathematik*. 7. vollständig überarbeitete und ergänzte Auflage. Frankfurt am Main: Verlag Harri Deutsch, 2008. ISBN: 978-3-8171-2007-9.
- [Bro16] Stan D. Brotherton. "Polycrystalline Silicon Thin Film Transistors (Poly-Si TFTs)". In: *Handbook of Visual Display Technology*. Hrsg. von Janglin Chen, Wayne Cranton und Mark Fihn. Cham: Springer International Publishing, 2016, S. 911–942. ISBN: 978-3-319-14346-0. DOI: 10.1007/978-3-319-14346-0\_48.
- [Bur21] Bureau International des Poids et Mesures. *SI Brochure: The International System of Units (SI). Units for photochemical and photobiological quantities*. Online. (Zugegriffen am 09.11.2021). BIPM, März 2021. URL: <https://www.bipm.org/documents/20126/41483022/SI-Brochure-9-App3-EN.pdf>.
- [Bur+13] Nino Burini, Ehsan Nadernejad, Jari Korhonen, Soren Forchhammer und Xiaolin Wu. "Modeling Power-Constrained Optimal Backlight Dimming for Color Displays". In: *Journal of Display Technology* 9.8 (Aug. 2013), S. 656–665. DOI: 10.1109/jdt.2013.2253544.
- [CR68] F. W. Campbell und J. G. Robson. "Application of fourier analysis to the visibility of gratings". In: *The Journal of Physiology* 197.3 (1968), S. 551–566. DOI: 10.1113/jphysiol.1968.sp008574.
- [Che+19] Chun-Chi Chen, Yan-Xue Wang, Young-Yuan Qiu, Gang Yu, Chung-Yi Chiu und Xin Zhang Bin Zhao. "55" High Contrast Ratio Panel Produced by Pixel Level Local Dimming Technology". In: *Proceedings of the International Display Workshops (IDW)* (Nov. 2019), S. 269–272. ISSN: 1883-2490/26/0269.
- [Che+17a] Hai-Wei Chen, Jiun-Haw Lee, Bo-Yen Lin, Stanley Chen und Shin-Tson Wu. "Liquid crystal display and organic light-emitting diode display: present status and future perspectives". In: *Light: Science & Applications* 7.3 (Dez. 2017), S. 17168–17168. DOI: 10.1038/lsa.2017.168.

- [Che+17b] Haiwei Chen, Guanjun Tan, Ming-Chun Li, Seok-Lyul Lee und Shin-Tson Wu. "Depolarization effect in liquid crystal displays". In: *Optics Express* 25.10 (Mai 2017), S. 11315. DOI: 10.1364/oe.25.011315.
- [Che+17c] Haiwei Chen, Ruidong Zhu, Ming-Chun Li, Seok-Lyul Lee und Shin-Tson Wu. "60-1: High Dynamic Range LCD with Pixel-level Local Dimming". In: *SID Symposium Digest of Technical Papers* 48.1 (Mai 2017), S. 890–893. DOI: 10.1002/sdtp.11771.
- [Che+17d] Haiwei Chen, Ruidong Zhu, Ming-Chun Li, Seok-Lyul Lee und Shin-Tson Wu. "Pixel-by-pixel local dimming for high-dynamic-range liquid crystal displays". In: *Optics Express* 25.3 (Jan. 2017), S. 1973. DOI: 10.1364/oe.25.001973.
- [Che+06] Hanfeng Chen, Junho Sung, Taehyeun Ha, Yungjun Park und Changwan Hong. "Backlight Local Dimming Algorithm for High Contrast LCD-TV". In: *Proc. of ASID*. (Zugegriffen am 06.11.2019). New Delhi, 2006, S. 168–171. URL: [http://www.iitk.ac.in/asid06/proceedings/papers/TC2\\_3-C.pdf](http://www.iitk.ac.in/asid06/proceedings/papers/TC2_3-C.pdf).
- [Che+09] Yu-Kuo Cheng, Yen-Hsing Lu, Chung-Hao Tien und Han-Ping Shieh. "Design and Evaluation of Light Spread Function for Area-Adaptive LCD System". In: *Journal of Display Technology* 5.2 (Feb. 2009), S. 66–71. ISSN: 1551-319X. DOI: 10.1109/jdt.2008.2004279.
- [Che+08] Yu-Kuo Cheng, Yen-Hsing Lu, Po-I. Lu, Chung-Hao Tien und Han-Ping D. Shieh. "64.3: Super-Gaussian Light Spread Function for High Dynamic Range Displays". In: *SID Symposium Digest of Technical Papers* 39 (Mai 2008). DOI: 10.1889/1.3069846.
- [CH17] Mina Choi und David M. Hoffman. "Efficacy of global dimming backlight and high-contrast liquid crystal panel for high-dynamic-range displays". In: *Journal of the Society for Information Display* 25.5 (2017), S. 283–294. DOI: 10.1002/jsid.549.
- [Col+09] Thomas H. Cormen (Dartmouth College), Charles E. Leiserson (MIT), Ronald L. Rivest (MIT) und Clifford Stein (Columbia University). *Introduction to Algorithms*. MIT Press Ltd, 31. Juli 2009. 1320 S. ISBN: 0-26-203384-4.
- [Cum19] Logal Cummins. *Local Dimming in Automotive Displays*. Online, TI Tech Days. (Zugegriffen am 19.01.2020). Okt. 2019. URL: <https://training.ti.com/sites/default/files/local-dimming-in-automotive-displays.pdf>.
- [Cum20a] Logal Cummins. *Implementing Automotive Displays with SerDes Daisy Chaining & Local Dimming Backlight Architecture*. Online, TI Tech Days. (Zugegriffen am 17.01.2022). 2020. URL: <https://www.ti.com/lit/ml/slyp694/slyp694.pdf>.
- [Cum20b] Logan Cummins. "An alternative to OLED with full-array local dimming in automotive displays". In: *SID Vehicle Displays and Interfaces* (Okt. 2020), S. 129–132.
- [Dan21] Daniel Research Group. *Average lifespan (replacement cycle length) of smartphones in the United States from 2014 to 2025*. In: Statista. (Zugegriffen am 16.12.2021). Okt. 2021. URL: <https://www.statista.com/statistics/619788/average-smartphone-life/>.
- [Den+18] Zhuo Deng, Binyi Zheng, Jianping Zheng, Ling Wu, Wenbin Yang, Zongwei Lin, Poping Shen und Junyi Li. "74-5: Late-News Paper: High Dynamic Range Incell LCD with Excellent Performance". In: *SID Symposium Digest of Technical Papers* 49.1 (Mai 2018), S. 996–998. DOI: 10.1002/sdtp.12215.

- [DG09] K. Diekmann und T. Gerloff. "Luminance homogeneity of OLED lighting tiles - A route to standardisation". In: *The OSA Direct Magazine* 2.9 (2009), S. 24–27.
- [DIN17a] DIN 5031. *Strahlungsphysik im optischen Bereich und Lichttechnik*. 2017. DOI: 10.31030/2700266.
- [DIN] DIN EN IEC 62087. *Audio-, Video- und verwandte Geräte - Messverfahren für die Leistungsaufnahme*.
- [DIN17b] DIN EN ISO 15008:2017-07. *Straßenfahrzeuge - Ergonomische Aspekte von Fahrerinformations- und Assistenzsystemen - Anforderungen und Bewertungsmethoden der visuellen Informationsdarstellung im Fahrzeug (ISO 15008:2017)*. Juli 2017. DOI: 10.31030/2585231.
- [Dis12] International Committee for Display Measurement (ICDM). *International Display Measurement Standard (IDMS)*. v1.03. Society for Information Display, Juni 2012. URL: <http://icdm-sid.org>.
- [Don+19] Hang Dong, Xin Mou, Guo Wei Chen und You Xiong Feng. "14.4: Driving to the Future—Flexible Active Matrix Organic Light Emitting Diode for Automotive Application". In: *SID Symposium Digest of Technical Papers* 50.S1 (Sep. 2019), S. 144–147. DOI: 10.1002/sdtp.13416.
- [Dud] "Gesichtswinkel" auf Duden online. Online. (Zugegriffen am 16.03.2020). URL: <https://www.duden.de/rechtschreibung/Gesichtswinkel>.
- [Dyb+05] Marc Dyble, Nadarajah Narendran, Andrew Bierman und Terence Klein. "Impact of dimming white LEDs: chromaticity shifts due to different dimming methods". In: *Fifth International Conference on Solid State Lighting*. Hrsg. von Ian T. Ferguson, John C. Carrano, Tsunemasa Taguchi und Ian E. Ashdown. SPIE, Aug. 2005. DOI: 10.1117/12.625924.
- [EIA01] EIA/TIA-644-A:2001-02. *Electrical Characteristics of Low Voltage Differential Signaling (LVDS) Interface Circuits*. Feb. 2001.
- [ELD] ELDIM S.A. *ELDIM UMaster*. Online. (Zugegriffen am 03.11.2019). URL: <http://eldim.biz/products/uniformity/umaster>.
- [Emm13] Jakob Emmel. "Novel backlighting techniques for high dynamic range displays". Diss. University of British Columbia, 2013. DOI: 10.14288/1.0085638.
- [FKL06] H.D. Foster, A.C. Krolnik und D.J. Lacey. "Assertion-Based Design". In: Springer US, 2006. ISBN: 9781402080289.
- [Gar08] Rags Gardner. *Eye to Eye Cameras and Vision*. Online. (Zugegriffen am 26.11.2021). 1. Aug. 2008. URL: <http://www.rags-int-inc.com/PhotoTechStuff/CameraEye/>.
- [Ger18] German Automotive OEM Working Group. *Display Specification for Automotive Application*. V5.1. 2018.
- [Ger] German Flat Panel Display Forum (DFF). *Display Measurement Specification (DMS) for Automotive-TFT LCDs*.
- [Ges14] Ralf Gessler. *Entwicklung Eingebetteter Systeme*. Springer Vieweg, 2014. ISBN: 978-3-8348-2080-8. DOI: 10.1007/978-3-8348-2080-8.
- [Gon08] Rafael Gonzalez. *Digital image processing*. Upper Saddle River, N.J.: Prentice Hall, 2008. ISBN: 978-0135-05267-9.

- [Gör08] Patrick Görn. "Transparente Elektronik für Aktiv-Matrix-Displays". Diss. Göttingen, 2008. ISBN: 978-3-86727-758-7.
- [GH07] Pierre de Greef und Hendriek Groot Hulze. "39.1: Adaptive Dimming and Boosting Backlight for LCD-TV Systems". In: *SID Symposium Digest of Technical Papers* 38.1 (Mai 2007), S. 1332–1335. DOI: 10.1889/1.2785558.
- [Grü+19] Michael Grüning, Julian Ritter, Maxim Schmidt und Chihao Xu. "22-3: 2400 Nits Edge-Lit LCD and Adaptive EOTF for HDR and Brilliant Images". In: *SID Symposium Digest of Technical Papers* 50.1 (2019), S. 307–310. DOI: 10.1002/sdtp.12917.
- [Grü+16] Michael Grüning, Maxim Schmidt, Daniel Schäfer und Chihao Xu. "Challenges and Methods for Local Dimming of Long-Edge LCD TVs". In: *Proceedings of the International Display Workshops (IDW)* (2016), S. 1339–1341.
- [Gu+20] Mingxia (Vincent) Gu, Ron Liu, Marc Albrecht, Jun Qi und Victor Yin. "10-4: Invited Paper: Taking High Dynamic Range Further: Apple Pro Display XDR". In: *SID Symposium Digest of Technical Papers* 51.1 (Aug. 2020), S. 126–129. DOI: 10.1002/sdtp.14593.
- [Hah10] H. Hahn. *Hellempfindlichkeitskurve*. Online. (Zugegriffen am 30.11.2021). Jan. 2010. URL: <https://de.wikipedia.org/wiki/Datei:EM-Spektrum.svg>.
- [Han18] Handelsblatt. *Grenzwerte für den CO<sub>2</sub>-Flottenausstoß bei Neuwagen von Automobilherstellern in der Europäischen Union im Zeitraum der Jahre 2015 bis 2030*. In: Statista. (Zugegriffen am 16.01.2020). Okt. 2018. URL: <https://de.statista.com/statistik/daten/studie/929825/umfrage/grenzwerte-fuer-den-co2-flottenausstoß-von-automobilherstellern-in-der-eu/>.
- [HSX16a] David M. Hoffman, Natalie Stepien und Wei Xiong. "59-2: Distinguished Paper: The Role of Local Dimming Density, Native Panel Contrast, and Glare Sources in the Visual Quality of HDR Displays". In: *SID Symposium Digest of Technical Papers* 47.1 (Mai 2016), S. 802–805. DOI: 10.1002/sdtp.10794.
- [HSX16b] David M. Hoffman, Natalie N. Stepien und Wei Xiong. "The importance of native panel contrast and local dimming density on perceived image quality of high dynamic range displays". In: *Journal of the Society for Information Display* 24.4 (2016), S. 216–228. DOI: 10.1002/jsid.416.
- [Hop12] Konrad Hoppe. *Das Weber-Fechner'sche Gesetz unter besonderer Berücksichtigung des Weber'schen Gesetzes*. Posthum veröffentlichtes Manuskript. Online. (Zugegriffen am 04.12.2021). Apr. 2012. URL: <https://elmer-herzig.de/wp-content/uploads/2012/03/Hoppe-posthum1.pdf>.
- [Hsi+20] En-Lin Hsiang, Qian Yang, Ziqian He, Junyu Zou und Shin-Tson Wu. "Halo effect in high-dynamic-range mini-LED backlit LCDs". In: *Optics Express* 28.24 (Nov. 2020), S. 36822–36837. DOI: 10.1364/oe.413133.
- [HLL08] Ching-Fu Hsu, Chih-Chang Lai und Jyun-Sian Li. "51.4: Backlight Power Reduction and Image Contrast Enhancement Using Adaptive Dimming for Global Backlight Applications". In: *SID Symposium Digest of Technical Papers* 39.1 (2008), S. 776–779. DOI: 10.1889/1.3069783.

- [Hua+19] Yuge Huang, Guanjun Tan, Fangwang Gou, Ming-Chun Li, Seok-Lyul Lee und Shin-Tson Wu. "Prospects and challenges of mini-LED and micro-LED displays". In: *Journal of the Society for Information Display* 27.7 (März 2019), S. 387–401. DOI: 10.1002/jsid.760.
- [Hub21] Andreas Huber. *Continental Shy Tech Display (2021): Touchscreen, Infotainment, Bildschirm, Dekor. Continental zeigt ein Display, das nur auftaucht, wenn man es braucht!* Hrsg. von Auto Bild. Online. (Zugegriffen am 18.01.2022). Nov. 2021. URL: <https://www.autobild.de/artikel/continental-shy-tech-display-2021-touchscreen-infotainment-bildschirm-dekor-20878629.html>.
- [Hub22] Andreas Huber. *Sony Vision-S 02 (2022): Prototyp, Elektroauto, Playstation. Playstation 5-Hersteller Sony zeigt als zweites Elektroauto ein SUV.* Hrsg. von Auto Bild. Online. (Zugegriffen am 18.01.2022). 6. Jan. 2022. URL: <https://www.autobild.de/artikel/sony-vision-s-02-2022-prototyp-elektroauto-playstation-21114779.html>.
- [Hd09] Hendriek Groot Hulze und Pierre deGreef. "50.2: Power Savings by Local Dimming on a LCD Panel with Side Lit Backlight". In: *SID Symposium Digest of Technical Papers* 40.1 (2009), S. 749–752. DOI: 10.1889/1.3256896.
- [IC 19] IC Knowledge LLC. *IC Cost and Price Model User Manual.* (Zugegriffen am 25.09.2020). 2019. URL: <https://icknowledge.com/products/icmodel.html>.
- [Ich+10] Hideki Ichioka, Katsuya Otoi, Kohji Fujiwara, Katsuteru Hashimoto, Hirofumi Murakami und Tomohiko Yamamoto. "50.3: Proposal of Evaluation Method for Local-Dimming Backlights". In: *SID Symposium Digest of Technical Papers* 41.1 (2010), S. 750–753. ISSN: 2168-0159. DOI: 10.1889/1.3500580.
- [IFP05] A. Iranli, H. Fatemi und M. Pedram. "HEBS: Histogram Equalization for Backlight Scaling". In: *Design, Automation and Test in Europe*. IEEE, 2005. DOI: 10.1109/date.2005.174.
- [ISO18] ISO 26262:2018. *Road vehicles – Functional safety*. Norm. 2018.
- [Jia18] Xingtong Jiang. "Modeling of OLED degradation for prediction and compensation of AMOLED aging artifacts". en. Diss. Universität des Saarlandes, 2018. DOI: 10.22028/D291-28013.
- [JVX15] Xingtong Jiang, Pascal Volkert und Chihao Xu. "Degradation Behavior of blue OLEDs". In: *Digest of Eurodisplay 2015* 1 (2015), S. 62. DOI: 10.1002/sdtp.10538.
- [JX21] Xingtong Jiang und Chihao Xu. "A unified OLED aging model combining three modeling approaches for extending AMOLED lifetime". In: *Journal of the Society for Information Display* 29.10 (Juni 2021), S. 768–784. DOI: 10.1002/jsid.1064.
- [Jo+18] Junho Jo, Jae Sung Park, Jae Woong Soh und Nam Ik Cho. "Convolutional Neural Network Driven Localized Backlight Dimming Strategy for Liquid Crystal Displays". In: *Proceedings of the International Display Workshops (IDW)* (2018), S. 1468–1470. ISSN: 1883-2490/25/1468.
- [Jon+99] James M. Jonza, Michael F. Weber, Andrew J. Ouderkirk und Carl A. Stover. "Polarizing beam-splitting optical component". US5962114. 1999. URL: <https://patents.google.com/patent/US5962114>.
- [Jun14] Tobias Jung. "Methoden zur Sicherstellung des visuellen Qualität in lokal gedimmten Backlightsystemen". Diss. Saarland University, 2014. DOI: 10.22028/D291-23034.

- [Jun+11] Tobias Jung, Marc Albrecht, Daniel Schäfer und Chihao Xu. "46.2: System Architecture and FPGA-Implementation of the SSC Local Dimming Processor for an Edge-Lit Serial TV". In: *SID Symposium Digest of Technical Papers* 42.1 (Juni 2011), S. 665–668. DOI: 10.1889/1.3621410.
- [Jun+13] Tobias Jung, Qiang Fu, Chihao Xu, Markus Kreuzer und Jan Bauer. "P.33: Correlation-Enhanced SSC Local Dimming Algorithm for Homogeneous Backlight Distributions of Edge-Lit LCDs". In: *SID Symposium Digest of Technical Papers* 44.1 (Juni 2013), S. 1110–1113. DOI: 10.1002/j.2168-0159.2013.tb06420.x.
- [Käl99] Kähil Käläntär. "34.2: Functional Light-Guide Plate for Backlight Unit". In: *SID Symposium Digest of Technical Papers* 30.1 (1999), S. 764. DOI: 10.1889/1.1834137.
- [Käl12] Kähil Käläntär. "A directional backlight with narrow angular luminance distribution for widening the viewing angle for an LCD with a front-surface light-scattering film". In: *Journal of the Society for Information Display* 20.3 (2012), S. 133. DOI: 10.1889/jsid20.3.133. URL: <https://sid.onlinelibrary.wiley.com/doi/abs/10.1889/JSID20.3.133>.
- [Kam21] Julian Kamasa. *Microchips: Small and demanded*. ETH Zurich, 2021. DOI: 10.3929/ethz-b-000517399.
- [KK09] Suk-ju Kang und Young Kim. "Image integrity-based gray-level error control for low power liquid crystal displays". In: *IEEE Transactions on Consumer Electronics* 55.4 (Nov. 2009), S. 2401–2406. DOI: 10.1109/tce.2009.5373816.
- [KZ19] KBA und Die Zeit. *Durchschnittliches Alter von Pkw in Deutschland in den Jahren 1960 bis 2019*. In: Statista. (Zugegriffen am 16.01.2020). Apr. 2019. URL: <https://de.statista.com/statistik/daten/studie/154506/umfrage/durchschnittliches-alter-von-pkw-in-deutschland/>.
- [KD06] Louis Kerofsky und Scott Daly. "26.2: Distinguished Paper: Brightness Preservation for LCD Backlight Reduction". In: *SID Symposium Digest of Technical Papers* 37.1 (2006), S. 1242. DOI: 10.1889/1.2433203.
- [Kik+19] Sho Kikuchi, Senshi Nasu, Takahiro Ishinabe und Hideo Fujikake. "Design of Mini-LED Backlight Using Reflective Mirror Dots with High Luminance Uniformity for Mobile LCDs". In: *Proceedings of the International Display Workshops (IDW)* (2019), S. 694–696. ISSN: 1883-2490/26/0694.
- [Kim+09] Seong-Eun Kim, Joo-Young An, Jong-Ju Hong, Tae Wook Lee, Chang Gone Kim und Woo-Jin Song. "How to reduce light leakage and clipping in local-dimming liquid-crystal displays". In: *Journal of The Society for Information Display - J SOC INF DISP* 17 (Dez. 2009). DOI: 10.1889/JSID17.12.1051.
- [KC66] T. I. Kirkpatrick und N. R. Clark. "Pert as an Aid to Logic Design". In: *IBM Journal of Research and Development* 10.2 (März 1966), S. 135–141. DOI: 10.1147/rd.102.0135.
- [Koe20] Martin Koers. *CO<sub>2</sub>-Regulierung bei Pkw und leichten Nutzfahrzeugen*. Online. (Zugegriffen am 04.03.2020). Verband der Automobilindustrie (VDA), 2020. URL: <https://www.vda.de/de/themen/umwelt-und-klima/co2-regulierung-bei-pkw-und-leichten-nfz/co2-regulierung-bei-pkw-und-leichten-nutzfahrzeugen.html>.

- [Kor+11] Jari Korhonen, Nino Burini, Søren Forchhammer und Jesper M. Pedersen. "Modeling LCD displays with local backlight dimming for image quality assessment". In: *Color Imaging XVI: Displaying, Processing, Hardcopy, and Applications*. Hrsg. von Reiner Eschbach, Gabriel G. Marcu und Alessandro Rizzi. SPIE, Jan. 2011. DOI: 10.1117/12.872375.
- [Kor06] B. H. Korte. *Combinatorial optimization : theory and algorithms*. Berlin New York: Springer, 2006. ISBN: 3-54-025684-9.
- [KSK18] D Koseoglu, Y S Sezer und K Karsli. "Colour gamut enhancement with remote light conversion mechanism". In: *Journal of Physics: Conference Series* 961 (Jan. 2018), S. 012010. DOI: 10.1088/1742-6596/961/1/012010.
- [Kra21] Alex Kraus. *Continental AG Showcases New Automobile Technology*. Hrsg. von Bloomberg. Online. (Zugegriffen am 11.11.2021). Nov. 2021. URL: <https://www.gettyimages.de/detail/nachrichtenfoto/shytech-display-activated-by-user-touch-is-displayed-nachrichtenfoto/1236181996>.
- [Kru+17] Kevin Kruse, Zachary Dolloff, David Rousseau und Kimberly Peiler. "Local Dimming LED Backlights for Automotive Displays". In: *Vehicle Displays and Interfaces Symposium*. Society for Information Display. Burton Manor Conference Center Livonia, Michigan, U.S.A., 2017, S. 391–409.
- [Lan08] Gerhard Lang. *Augenheilkunde : Verstehen - Lernen - Anwenden*. Stuttgart: Thieme, 2008. ISBN: 978-3-13-102834-1.
- [LST09] You-Ching Lee, Cheng-En Shie und Din-Chang Tseng. "LCD Mura Detection Based on Accumulated Differences and Multi-resolution Background Subtraction". In: *2009 Fifth International Conference on Image and Graphics*. IEEE, Sep. 2009. DOI: 10.1109/icig.2009.183.
- [Lev+11] Leonard A. Levin, Siv F. E. Nilsson, James Ver Hoeve, Samuel M. Wu, Paul L. Kaufman und Albert Alm. *Adler's Physiology of the Eye*. Elsevier Health Science, 1. Apr. 2011. ISBN: 978-0-323-05714-1.
- [Li+07] Feng Li, Xiaofan Feng, Ibrahim Sezan und Scott Daly. "Two approaches to derive LED driving signals for high-dynamic-range LCD backlights". In: *Journal of the Society for Information Display* 15.12 (2007), S. 989. DOI: 10.1889/1.2825109.
- [LNW10] Yubo Li, Brent Nelson und Michael Wirthlin. "Synchronization Techniques for Crossing Multiple Clock Domains in FPGA-Based TMR Circuits". In: *IEEE Transactions on Nuclear Science* (Dez. 2010). DOI: 10.1109/tns.2010.2086075.
- [LH10] L. Liao und Y. Huang. "Blur-Mask Approach for Real-Time Calculation of Light Spreading Function (LSF) on Spatial Modulated High Dynamic Range LCDs". In: *Journal of Display Technology* 6.4 (Apr. 2010), S. 121–127. ISSN: 1551-319X. DOI: 10.1109/JDT.2009.2035826.
- [Lin+07] Fang-Cheng Lin, Cheng-Yu Liao, Lin-Yao Liao, Yi-Pai Huang, Han-Ping D. Shieh, Po-Jen Tsai, Te-Mei Wang und Yao-Jen Hsieh. "39.4: Inverse of Mapping Function (IMF) Method for Image Quality Enhancement of High Dynamic Range LCD TVs". In: 38.1 (Mai 2007), S. 1343–1346. DOI: 10.1889/1.2785561.

- [Liu+19] Bin Liu, Quansheng Liu, Yongyuan Qiu Jia Li, Junling Liu, Yong Yang, Hongyuan Xu, Juncheng Xiao, Feng Zhu, Hang Zhou und Xin Zhang. "An Active Matrix Mini-LEDs Backlight based a-Si TFT". In: *Proceedings of the International Display Workshops (IDW)* (Nov. 2019), S. 652–654. ISSN: 1883-2490/26/0652.
- [LCT07] Yen-Hsing Lu, Yu-Kuo Cheng und Chung-Hao Tien. "P-69: A Localized Partition Approach for High-Dynamic-Range Display". In: *SID Symposium Digest of Technical Papers* 38.1 (2007), S. 449–452. DOI: 10.1889/1.2785330.
- [Lue01] Ernst Lueder. *Liquid crystal displays : addressing schemes and electro-optical effects*. Chichester New York: John Wiley und Sons, 2001. ISBN: 978-0-4714-9029-6.
- [Mar18] IHS Markit. Online. (Zugegriffen am 21.05.2020). 2018. URL: <https://autotechinsight.ihsmarket.com/>.
- [Mas+19] Takeshi Masuda, Hisashi Watanabe, Youzou Kyokane, Hirotohi Yasunaga, Hidekazu Miyata, Masafumi Yashiki, Tomoyuki Nara und Takeshi Ishida. "28-3: Mini-LED Backlight for HDR Compatible Mobile Displays". In: *SID Symposium Digest of Technical Papers* 50.1 (Mai 2019), S. 390–393. DOI: 10.1002/sdtp.12939.
- [Mat09] Matt. *Elektromagnetisches Spektrum*. Online. (Zugegriffen am 30.11.2021). Juli 2009. URL: <https://de.wikipedia.org/wiki/Datei:V-lambda-phot-scot.svg>.
- [MNS19] Ulf Meiners, Takuya Noichi und Shintaro Suhara. "Advanced Direct-lit Light Panel Technology for Automotive Displays". In: *electronic displays Conference proceedings* ISBN 978-3-645-50183-5 Digitale Proceedings (Feb. 2019).
- [Mek18] Meko (displaydaily.com). *Tianma Impressive as Usual*. Online. (Zugegriffen am 20.01.2020). 2018. URL: <https://www.displaydaily.com/?view=article&id=62270:tianma-impressive-as-usual>.
- [Mog+19] Felix Mogge, Thomas Schlick, Florian Daniel, Ken Oliver Fritz und Christof Söndermann. *Global Automotive Supplier Study 2019 - After years of excellent growth, the market environment is getting difficult*. Lazard, Roland Berger, Aug. 2019.
- [MLV08] Remco Muijs, Erno Langendijk und Frank Vossen. "64.2: Spatio-temporally Consistent Video Processing for Local Backlight Dimming". In: *SID Symposium Digest of Technical Papers* 39.1 (2008), S. 979. DOI: 10.1889/1.3069844.
- [MH12] Michael J. Murdoch und Ingrid E. J. Heynderickx. "Veiling glare and perceived black in high dynamic range displays". In: *Journal of the Optical Society of America A* 29.4 (März 2012), S. 559. DOI: 10.1364/josaa.29.000559.
- [Mur64] B. T. Murphy. "Cost-size optima of monolithic integrated circuits". In: *Proceedings of the IEEE* 52.12 (1964), S. 1537–1545.
- [OLE21] OLED-info.com. *Mercedes unveiled its 2022 EQE-350 EV, with internal OLED displays*. Online. (Zugegriffen am 12.09.2021). 8. Sep. 2021. URL: <https://www.oled-info.com/mercedes-unveiled-its-2022-eqe-350-ev-internal-oled-displays>.
- [OSR] OSRAM Opto Semiconductors. *Mini TOPLED LCW MVSG.EC. Small size high-flux LED for slim designs*. Online. Version 1.3. (Zugegriffen am 10.01.2022). URL: [https://dammedia.osram.info/media/resource/hires/osram-dam-5584202/LCW%20MVSG.EC\\_EN.pdf](https://dammedia.osram.info/media/resource/hires/osram-dam-5584202/LCW%20MVSG.EC_EN.pdf).

- [Ped+02] Frank Pedrotti, Leno Pedrotti, Werner Bausch und Hartmut Schmidt. *Optik für Ingenieure Grundlagen. Grundlagen*. 3. Aufl. Springer-Verlag Heidelberg, 2002. ISBN: 3-54-067379-2.
- [Pho] JADAK \ Photoresearch. *SpectraScan® Spectroradiometer PR-740*. Online. (Zugegriffen am 21.10.2021). URL: <https://www.jadaktech.com/products/photo-research/spectrascan-pr-740/>.
- [Poy03] Charles Poynton. *Digital video and HDTV : algorithms and interfaces*. Amsterdam Boston: Morgan Kaufmann Publishers, 2003. ISBN: 978-1-55860-792-7.
- [Pre+07] William H. Press, Saul A. Teukolsky, William T. Vetterling und Brian P. Flannery. *Numerical Recipes 3rd Edition*. Bd. 3rd Revised edition. Cambridge University Press, 2007. ISBN: 978-0-521-88068-8.
- [Qib+15] Feng Qibin, He Huijie, Han Dong, Zhang Lei und Lv Guoqiang. "Image-classification-based global dimming algorithm for LED backlights in LCDs". In: *Optics & Laser Technology* 70 (2015), S. 106–111. ISSN: 0030-3992. DOI: <https://doi.org/10.1016/j.optlastec.2014.12.021>.
- [Ras16] Torben Rasmussen. *flatpanelshd. OLED vs. LCD*. Online. (Zugegriffen am 12.12.2021). 23. Sep. 2016. URL: <https://www.flatpanelshd.com/focus.php?subaction=showfull&id=1474618766>.
- [Raz01] Behzad Razavi. *Design of analog CMOS integrated circuits*. Boston, MA: McGraw-Hill, 2001. ISBN: 0071188398.
- [RDH10] Erik Reinhard, Paul Debevec und Wolfgang Heidrich. *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting*. Morgan Kaufmann Publishers, 31. Mai 2010. 650 S. ISBN: 978-0-12-374914-7.
- [Rie14] Boris Riedel. "Effizienzsteigerung in organischen Leuchtdioden". Diss. Karlsruher Institut für Technologie, 22. Aug. 2014. 214 S. ISBN: 3866447000.
- [Sch17] Daniel Schäfer. "Herausforderungen und Qualitätssicherung beim lokalen Dimmen von LC-Displays mit seitlich angeordneten LEDs". Diss. Universität des Saarlandes, 2017. DOI: 10.22028/D291-27127.
- [Sch+13] Daniel Schäfer, Tobias Jung, Michael Krause und Chihao Xu. "Efficient Local Dimming Algorithm for LCDs with Single-Edge LED Backlight". In: *Proceedings of the International Display Workshops (IDW)*. 2013, S. 1485–1488.
- [SJM12] Daniel Schäfer, Tobias Jung und Chihao Xu. "36.2: Enhanced Local Pixel Compensation with Clipping Suppression and Global Luminance Preservation". In: *SID Symposium Digest of Technical Papers* 43.1 (Juni 2012), S. 473–476. DOI: 10.1002/j.2168-0159.2012.tb05820.x.
- [Sch+16a] Daniel Schäfer, Yuxin Zhang, Michael Grüning, Chihao Xu, Shunming Huang, Weidong Liu und Jianwei Cao. "Brilliant Images and Saturated Colors for 4K Edge-Lit LED TVs generated by an Efficient Versatile SSC Local Dimming Processor". In: *SID Symposium Digest of Technical Papers* 47 (2016), S. 211–214. DOI: 10.1002/sdtp.10639.
- [SAX20] Maxim Schmidt, Ramazan Ayasli und Chihao Xu. "Customized Local Dimming Algorithm and BLU for Automotive Application towards Low Power Consumption and High Visual Quality". In: *SID Vehicle Displays and Interfaces Paper No. 5.2* (Okt. 2020), S. 93–96.

- [Sch+19b] Maxim Schmidt, Michael Grüning, Julian Ritter, Andreas Hudak und Chihao Xu. "Impact of high-resolution matrix backlight on local-dimming performance and its characterization". In: *Journal of Information Display* 20.02 (2019), S. 95–104. ISSN: 2158-1606. DOI: 10.1080/15980316.2019.1606121.
- [Sch+17] Maxim Schmidt, Michael Grüning, Daniel Schäfer und Chihao Xu. "Efficient Modeling of LED Crosstalk of a Matrix Backlight Unit". In: *Proceedings of the International Display Workshops (IDW)* (Dez. 2017). ISSN-L 1883-2490/24/1457, S. 1457–1460.
- [SRX19] Maxim Schmidt, Julian Ritter und Chihao Xu. "Optimizing LSF Shape for Robust and Uniform Backlighting of Automotive Displays with Direct-Lit Local-Dimming". In: *Proceedings of the International Display Workshops (IDW)* (Nov. 2019), S. 1554–1557. ISSN: 1883-2490/26/1554. DOI: 10.36463/idw.2019.1554.
- [SSX15] Maxim Schmidt, Daniel Schäfer und Chihao Xu. "P-77: Increased Power Saving by Combining Global and Local Dimming for Edge-LED LCDs". In: *SID Symposium Digest of Technical Papers* 46 (Mai 2015), S. 1431–1434. DOI: 10.1002/sdtp.10002.
- [SSX16] Maxim Schmidt, Daniel Schäfer und Chihao Xu. "Verfahren und Vorrichtung zum Dimmen einer Flüssigkristallanzeige mit Hintergrundbeleuchtung". DE102016209222A1. Maxim Schmidt und X-Motive GmbH. 27. Mai 2016.
- [SX19] Maxim Schmidt und Chihao Xu. "Robust BLU Design by Optimizing LSF Shape for Automotive Applications". In: *SID Mid Europe Chapter 2019*. SID Mid Europe Chapter 2019. Eindhoven, The Netherlands, Juli 2019.
- [Sch+18] Maxim Schmidt, Chihao Xu, Michael Grüning, Julian Ritter und Andreas Hudak. "Design and Evaluation of High Resolution Matrix Backlight for Excellent Local-Dimming Results and Uniformity of LCDs". In: *IMID Digest* (2018).
- [Sch+16b] Maxim Schmidt, Chihao Xu, Torsten Lahr und Kai Hohmann. "Adaptive Global Dimming Based on Image Analysis for Power Saving and Preserving Brilliance of Automotive Displays". In: *electronic displays Conference proceedings* ISBN 978-3-645-50160-6 elektronische Tagungsunterlagen auf CD-ROM (Feb. 2016).
- [Sch06] Jörg Schmittler. "SaarCOR : a hardware architecture for real-time ray tracing". Diss. Universität des Saarlandes, 2006. DOI: 10.22028/D291-25876.
- [See+04] Helge Seetzen, Wolfgang Heidrich, Wolfgang Stuerzlinger, Greg Ward, Lorne Whitehead, Matthew Trentacoste, Abhijeet Ghosh und Andrejs Vorozcovs. "High Dynamic Range Display Systems". In: *ACM Trans. Graph.* 23.3 (Aug. 2004), S. 760–768. ISSN: 0730-0301. DOI: 10.1145/1015706.1015797.
- [SWW03] Helge Seetzen, Lorne A. Whitehead und Greg Ward. "54.2: A High Dynamic Range Display Using Low and High Resolution Modulators". In: *SID Symposium Digest of Technical Papers* 34.1 (2003), S. 1450–1453. DOI: <https://doi.org/10.1889/1.1832558>. URL: <https://sid.onlinelibrary.wiley.com/doi/abs/10.1889/1.1832558>.
- [SA14] Mark Segal und Kurt Akeley. *The OpenGL Graphics System. A Specification (Version 4.4 (Core Profile))*. Hrsg. von Chris Frazier, Jon Leech und Pat Brown. (Zugegriffen am 17.09.2021). Khronos Group, 19. Apr. 2014. URL: <https://www.khronos.org/registry/OpenGL/specs/gl/glspec44.core.pdf>.

- [Shi+18] Xuefeng Shi, Jianqun Chen, Liting Fang, Ling Wu, Poping Shen und Junyi Li. "Advanced RGBW Algorithm for High Image Quality with Low Power Consumption". In: *Proceedings of the International Display Workshops (IDW)* (2018).
- [SSM08] Tomokazu Shiga, Sho Shimizukawa und Shigeo Mikoshiba. "Power savings and enhancement of gray-scale capability of LCD TVs with an adaptive dimming technique". In: *Journal of the Society for Information Display* 16.2 (2008), S. 311. DOI: 10.1889/1.2841865.
- [Shi+06] T. Shirai, S. Shimizukawa, T. Shiga, S. Mikoshiba und K. Kälantär. "44.4: RGB-LED Backlights for LCD- TVs with 0D, 1D, and 2D Adaptive Dimming". In: *SID Symposium Digest of Technical Papers* 37.1 (2006), S. 1520. DOI: 10.1889/1.2433283.
- [SWF12] Xiao Shu, Xiaolin Wu und Søren Forchhammer. "Optimal Local Dimming for LC Image Formation With Controllable Backlighting". In: *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society* 22 (Aug. 2012). DOI: 10.1109/TIP.2012.2211371.
- [Sta05] Standard Panel Working Group (SPWG). *SPWG Notebook Panel Specification*. v3.5. März 2005.
- [SG06] Jeroen Stessen und J G. R. van Mourik. "26.4: Algorithm for Contrast Reserve, Backlight Dimming, and Backlight Boosting on LCD". In: *SID Symposium Digest of Technical Papers* 37 (Juni 2006). DOI: 10.1889/1.2433205.
- [Ste57] S. S. Stevens. "On the psychophysical law." In: *Psychological Review* 64.3 (1957), S. 153–181. DOI: 10.1037/h0046162.
- [Sto19] Dale Stoltzka. "An Analysis of the DisplayHDR Compliance Specification and Its DisplayHDR True Black Specification Extension". In: *Information Display* 35.3 (Mai 2019), S. 6–11. DOI: 10.1002/msid.1031.
- [Sun12] Jian Sun. "Pulse-Width Modulation". In: *Dynamics and Control of Switched Electronic Systems: Advanced Perspectives for Modeling, Simulation and Control of Power Converters*. Hrsg. von Francesco Vasca und Luigi Iannelli. London: Springer London, 2012, S. 25–61. ISBN: 978-1-4471-2885-4. DOI: 10.1007/978-1-4471-2885-4\_2.
- [Swi+06] S. Swinkels, R. Muijs, E. Langendijk und F. Vossen. "Effect of Backlight Segmentation on Perceived Image Quality for HDR Displays". In: *Proceedings of the International Display Workshops (IDW)* (2006).
- [Sys18] Radiant Vision Systems. *Testing for Black Mura Gradient per German Automotive (DFF) Standards*. Online. (Zugegriffen am 25.06.2021). Juni 2018. URL: [https://www.radiantvisionsystems.eu/fileadmin/user\\_upload/Learn/8\\_Tech\\_Notes/Radiant\\_TECH\\_Black-Mura-Gradient\\_EN\\_A4.pdf](https://www.radiantvisionsystems.eu/fileadmin/user_upload/Learn/8_Tech_Notes/Radiant_TECH_Black-Mura-Gradient_EN_A4.pdf).
- [Tan+18] Guanjun Tan, Yuge Huang, Ming-Chun Li, Seok-Lyul Lee und Shin-Tson Wu. "High dynamic range liquid crystal displays with a mini-LED backlight". In: *Opt. Express* 26.13 (Juni 2018), S. 16572–16584. DOI: 10.1364/OE.26.016572.
- [Tex14] Texas Instruments. *TLC5955 48-Channel*. (Zugegriffen am 28.03.2020). März 2014. URL: <https://www.ti.com/lit/ds/symlink/tlc5955.pdf>.
- [Tex16] Texas Instruments. *TFP401x TI PanelBus™ Digital Receiver*. Rev. G. (Zugegriffen am 23.03.2020). Mai 2016. URL: <https://www.ti.com/lit/gpn/TFP401>.

- [Tex18] Texas Instruments. *TI Designs: TIDA-020001 Automotive 144-Zone Local Dimming Backlight Reference Design*. Online. (Zugegriffen am 21.01.2020). Okt. 2018. URL: <http://www.ti.com/lit/ug/tiduea1/tiduea1.pdf>.
- [Tex20] Texas Instruments. *DS90UH948-Q1 2K FPD-Link III to OpenLDI Deserializer With HDCP*. (Zugegriffen am 17.12.2021). Dez. 2020. URL: <https://www.ti.com/lit/ds/symlink/ds90uh948-q1.pdf>.
- [Tin+19] Chin-Lung Ting, Chung-Kuang Wei, Ker-Yih Kao, Li-Wei Mau, Ho-Tien Chen und Minoru Shibasaki. "Active Matrix Driving mini-LED Device". In: *Proceedings of the International Display Workshops (IDW)* (Nov. 2019), S. 425–428. ISSN: 1883-2490/26/0425.
- [Toy04] Toyota. *PRIUS 2004 Owner Manual*. 2004.
- [Tsu17] Takatoshi Tsujimura. *Oled Display Fundamentals and Applications*. WILEY, 3. Apr. 2017. 320 S. ISBN: 1119187311.
- [Vid06] Video Electronics Standards Association. *VESA ENHANCED EXTENDED DISPLAY IDENTIFICATION DATA STANDARD*. Newark CA, USA, Sep. 2006.
- [Vid13] Video Electronics Standards Association. *VESA and Industry Standards and Guidelines for Computer Display Monitor Timing (DMT)*. Version 1.0, Rev. 13. Newark, CA, USA, Feb. 2013.
- [VX15] P. Volkert und C. Xu. "Compensation of OLED I-V-Drift for Suppressing Image Sticking in a Digital AMOLED Display Module". In: *SID Digest 1* (2015), S. 258–261.
- [Vol17] Pascal Volkert. "Digital gesteuertes AMOLED-System : Konzeption, Entwicklung und Realisierung". Diss. Universität des Saarlandes, 2017. DOI: 10.22028/D291-26835.
- [WY19] Preeti Wadhvani und Shubhangi Yadav. *Automotive Digital Instrument Cluster Market Size by Display Type (LCD, TFT-LCD, OLED), By Display Size (5-8-inch, 9-11 inch, >12 inch), Industry Analysis Report, Regional Outlook, Growth Potential, Competitive Market Share & Forecast, 2019 – 2025*. Global Market Insights, Inc. (Zugegriffen am 25.09.2020). Okt. 2019. URL: <https://www.gminsights.com/industry-analysis/automotive-digital-instrument-cluster-market>.
- [Wan+17] Jiang Wang, Hao Wang, Xiaohua Li, Qifeng Sun, Kai Jia und Subing Zhang. "Key Subject Evaluation Factors of HDR Image Quality Based on LCD and OLED Comparison". In: *Proceedings of the International Display Workshops (IDW)* (2017), S. 1032–1035.
- [War00] Warren J. Smith. *Modern Optical Engineering - The Design of Optical Systems*. Third Edition. Two Penn Plaza, New York, NY 10121-2298, USA: McGraw-Hill, 2000. ISBN: 0-07-136360-2.
- [Web+96] Michael F. Weber, Jr. Olester Benson, Jr. Sanford Cobb, James M. Jonza, Andrew J. Ouderkirk, David L. Wordman und Carl A. Stover. "Display with reflective polarizer and randomizing cavity". US-Pat. US6025897 (St. Paul, Minn). 1996. URL: <https://patentimages.storage.googleapis.com/11/43/ec/f4a89b3bb77211/US6025897.pdf>.
- [Wei+18] Chang Wei-Fu, Li Ching-Lung, Tsao Fu-Hsiang, Zhang Yi, Yang Sung-Hsin, Chen Hung-Hsiang, Su Ho-Ming und Tseng Wen-Tse. "An Automotive Grade Local Dimming Display System". In: *Proceedings of the International Display Workshops (IDW)* (2018), S. 807–810. ISSN: 1883-2490/25/0807.
- [Wei62] Paul Weimer. "The TFT A New Thin-Film Transistor". In: *Proceedings of the IRE* 50.6 (Juni 1962), S. 1462–1469. DOI: 10.1109/jrproc.1962.288190.

- [Wei16] Paul Weindorf. "36-1: Automotive OLED Luminance Consumption Control Methods". In: *SID Symposium Digest of Technical Papers* 47.1 (Mai 2016), S. 456–459. DOI: 10.1002/sdtp.10706.
- [Wil15] Mark James Burge Wilhelm Burger. *Digitale Bildverarbeitung*. Springer-Verlag GmbH, 4. Mai 2015. ISBN: 978-3-642-04603-2.
- [Xia+19] Juncheng Xiao, Jiayang Fei, Hongyuan Xu, Yongyuan Qiu, Quansheng Liu, Yong Yang, Junling Liu, Jiaqing Zhuang, Chunming Liu, Daobing Hu und Xin Zhang. "75-inch LCD Displays with AM MiniLED Local Dimming Backlight Units on Glass". In: *Proceedings of the International Display Workshops (IDW)* (Nov. 2019), S. 417–420. ISSN: 1883-2490/26/0417.
- [Xila] Xilinx. *The Ultimate Guide to FPGA Architecture*. Online. (Zugegriffen am 06.01.2022). URL: <https://hardwarebee.com/the-ultimate-guide-to-fpga-architecture/>.
- [Xil11] Xilinx DS160. *Spartan-6 Family Overview*. v2.0. (Zugegriffen am 21.03.2020). 25. Okt. 2011. URL: [https://www.xilinx.com/content/dam/xilinx/support/documentation/data\\_sheets/ds160.pdf](https://www.xilinx.com/content/dam/xilinx/support/documentation/data_sheets/ds160.pdf).
- [Xil18] Xilinx DS180. *7 Series FPGAs Data Sheet: Overview*. v2.6. (Zugegriffen am 21.03.2020). Feb. 2018. URL: [https://www.xilinx.com/support/documentation/data\\_sheets/ds180\\_7Series\\_Overview.pdf](https://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf).
- [Xil17] Xilinx DS197. *XA Artix-7 FPGAs Data Sheet: Overview*. v1.3. (Zugegriffen am 21.03.2020). Nov. 2017. URL: [https://www.xilinx.com/support/documentation/data\\_sheets/ds197-xa-artix7-overview.pdf](https://www.xilinx.com/support/documentation/data_sheets/ds197-xa-artix7-overview.pdf).
- [Xilb] Xilinx UG474. *7 Series FPGAs Configurable Logic Block*. v1.8. (Zugegriffen am 06.01.2022). URL: [https://www.xilinx.com/support/documentation/user\\_guides/ug474\\_7Series\\_CLB.pdf](https://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf).
- [XD12] C. Xu und Y. Dobrev. "Digital Driving Method Utilizing Image Decomposition for Nonuniformity Compensation of AMOLED Displays". In: *Digest of IMID 2012* 63 (2012). IMID, S. 475–476.
- [XAJ12] Chihao Xu, Marc Albrecht und Tobias Jung. "Dimming of LED LCD Backlights". In: *Handbook of Visual Display Technology*. Hrsg. von Janglin Chen, Wayne Cranton und Mark Fihn. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, S. 567–574. ISBN: 978-3-540-79567-4. DOI: 10.1007/978-3-540-79567-4\_43.
- [XAJ16] Chihao Xu, Marc Albrecht und Tobias Jung. "Intelligent Control of LED LCD Backlights". In: *Handbook of Visual Display Technology*. Hrsg. von Janglin Chen, Wayne Cranton und Mark Fihn. Cham: Springer International Publishing, 2016, S. 817–826. ISBN: 978-3-319-14346-0. DOI: 10.1007/978-3-319-14346-0\_43.
- [XAS22] Chihao Xu, Ramazan Ayasli und Maxim Schmidt. "Optimized Algorithm and BLU for Local Dimming of Automotive Displays". In: *International Conference on Display Technology (ICDT)* (Apr. 2022). Paper akzeptiert.
- [XKA13] Chihao Xu, Andreas Karrenbauer und Mark Albrecht. "Method, system and apparatus for power saving backlight". US-Pat. US8421741B2. Eingereicht am 11.09.2009. 16. Apr. 2013. URL: <https://patentimages.storage.googleapis.com/75/07/68/771789429f0f45/US8421741.pdf>.

- [Xu+08] Chihao Xu, Andreas Karrenbauer, Kian Min Soh und Cosmin Codrea. "Consecutive multiline addressing: A scheme for addressing PMOLEDs". In: *Journal of the Society for Information Display* 16.2 (2008), S. 211. DOI: 10.1889/1.2841853.
- [Xu+19] Chihao Xu, Daniel Schäfer, Mark Albrecht und Tobias Jung. "Scheme for adjusting image data of LCD (liquid crystal display) by aid of local darkened backlight". CN000104637455 (China). Eingereicht am 15.11.2013. Juli 2019.
- [Xu+18a] Chihao Xu, Maxim Schmidt, Michael Grüning und Xingtong Jiang. "Image Processing for Enhancing Display Performance". In: *IMID Digest* (2018).
- [Xu+18b] Chihao Xu, Maxim Schmidt, Torsten Lahr und Markus Weber. "Dynamic Backlights for Automotive LCDs". In: *Information Display* (2018). DOI: 10.1002/j.2637-496x.2018.tb01099.x.
- [Yeo+08] Dong-Min Yeo, Yong-hoon Kwon, Eui-Jeong Kang, Se-Ki Park, Byungchoon Yang, Gicherl Kim und Taeseok Jang. "64.4: Smart Algorithms for Local Dimming LED Backlight". In: *SID Symposium Digest of Technical Papers* 39.1 (2008), S. 986–989. DOI: 10.1889/1.3069847.
- [You97] Matt Young. "Die elektromagnetische Theorie des Lichts und Polarisierungseffekte". In: *Optik, Laser, Wellenleiter*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, S. 253–281. ISBN: 978-3-642-60369-3. DOI: 10.1007/978-3-642-60369-3\_9.
- [Zha+19] Tao Zhang, Yifei Wang, Hongying Wu, Meng Li und Zhichun Lei. "High-performance local-dimming algorithm based on image characteristic and logarithmic function". In: *Journal of the Society for Information Display* 27.2 (2019), S. 85–100. DOI: <https://doi.org/10.1002/jsid.740>.
- [Zha+09] W. Zhang, M. Chen, Weihong Niu und Danding Huang. "LED Control signal extraction by using Multiple Representative Values". In: *Proceedings of the International Display Workshops (IDW)* (2009), S. 1519–1522.
- [Zhe+19a] Binyi Zheng, Zhuo Deng, Jianping Zheng, Ling Wu, Wenbin Yang, Zongwei Lin, Hailiang Wang, Poping Shen und Junyi Li. "41-2: Invited Paper: An Advanced High-Dynamic-Range LCD for Smartphones". In: *SID Symposium Digest of Technical Papers* 50.1 (Mai 2019), S. 566–568. DOI: 10.1002/sdtp.12983.
- [Zhe+19b] Jianping Zheng, Zhuo Deng, Ling Wu, Poping Shen, Junyi Li und Jianmou Huang. "A new generation of HDR display with super multi-zones mini LED". In: *Proceedings of the International Display Workshops (IDW)* (Nov. 2019). ISSN: 1883-2490/26/0645.

## Graue Literatur

- [Jos20] Matthias Jost. "Distributed Pixelorder for Parallel Execution of the SSC Algorithm Meeting Real Time Requirements of a High Resolution BLU". Bachelorarbeit. Universität des Saarlandes, Lehrstuhl für Mikroelektronik, Juni 2020.
- [Kar17] Artur Karcher. "Vergleich und FPGA-Implementierung von Algorithmen zur adaptiven Kontrastverbesserung für Videos im Consumer-Bereich". Masterarbeit. Universität des Saarlandes, Lehrstuhl für Mikroelektronik, Feb. 2017.
- [Sch13] Maxim Schmidt. "Untersuchung des Einsparpotentials im SSC Local-Dimming-Algorithmus für Office Anwendungen und mobile Endgeräte anhand geeigneter Klassifizierung des Bildinhalts auf Condenserebene". Masterarbeit. Universität des Saarlandes, Lehrstuhl für Mikroelektronik, Okt. 2013.
- [Sch+19a] Maxim Schmidt, Julian Bürner, Tim Goll, Daniel Schäfer und Chihao Xu. *Praktikum Mikroelektronik: Development of an FPGA-based Dimming-Algorithm for Reduction of the Power Consumption of an LC-Display - An Introduction to Digital Design with Verilog -*. Universität des Saarlandes, Lehrstuhl für Mikroelektronik. 2019.
- [Wer16] Simon Werner. "Optimierung des Einsparpotentials eines GPU-basierten Global-Dimming-Algorithmus für Flüssigkristallanzeigen in Automobilanwendungen". Masterarbeit. Universität des Saarlandes, Lehrstuhl für Mikroelektronik, Nov. 2016.