Non-Disruptive Use of Light Fields in Image and Video Processing

A dissertation submitted towards the degree

Doctor of Engineering (Dr.-Ing.)

of the Faculty of Mathematics and Computer Science

of Saarland University

by

Harini Priyadarshini Hariharan

Saarbrücken, 2022

Tag des Kolloquiums:	23. February 2022
Dekan:	Prof. Dr. Thomas Schuster
Prüfungsausschuss:	
Vorsitzende:	Prof. Dr. Volkhard Helms
Berichtserstatter:	Prof. Dr. Thorsten Herfet
	Prof. Dr. Joachim Weickert
Akademischer Mitarbeiter:	Dr. Michelle Carnell

Abstract

Non-Disruptive Use of Light Fields in Image and Video Processing

In the age of computational imaging, cameras capture not only an image but also data. This captured additional data can be best used for photo-realistic renderings facilitating numerous post-processing possibilities such as perspective shift, depth scaling, digital refocus, 3D reconstruction, and much more. In computational photography, the light field imaging technology captures the complete volumetric information of a scene. This technology has the highest potential to accelerate immersive experiences towards close-to-reality. It has gained significance in both commercial and research domains. However, due to lack of coding and storage formats and also the incompatibility of the tools to process and enable the data, light fields are not exploited to its full potential. This dissertation approaches the integration of light field data to image and video processing. Towards this goal, the representation of light fields using advanced file formats designed for 2D image assemblies to facilitate asset re-usability and interoperability between applications and devices is addressed. The novel 5D light field acquisition and the on-going research on coding frameworks are presented. Multiple techniques for optimised sequencing of light field data are also proposed. As light fields contain complete 3D information of a scene, large amounts of data is captured and is highly redundant in nature. Hence, by pre-processing the data using the proposed approaches, excellent coding performance can be achieved.

Zusammenfassung

Non-Disruptive Use of Light Fields in Image and Video Processing

Im Zeitalter der computergestützten Bildgebung erfassen Kameras nicht mehr nur ein Bild, sondern vielmehr auch Daten. Diese erfassten Zusatzdaten lassen sich optimal für fotorealistische Renderings nutzen und erlauben zahlreiche Nachbearbeitungsmöglichkeiten, wie Perspektivwechsel, Tiefenskalierung, digitale Nachfokussierung, 3D-Rekonstruktion und vieles mehr. In der computergestützten Fotografie erfasst die Lichtfeld-Abbildungstechnologie die vollständige volumetrische Information einer Szene. Diese Technologie bietet dabei das größte Potenzial, immersive Erlebnisse zu mehr Realitätsnähe zu beschleunigen. Deshalb gewinnt sie sowohl im kommerziellen Sektor als auch im Forschungsbereich zunehmend an Bedeutung. Aufgrund fehlender Kompressions- und Speicherformate sowie der Inkompatibilität der Werkzeuge zur Verarbeitung und Freigabe der Daten, wird das Potenzial der Lichtfelder nicht voll ausgeschöpft. Diese Dissertation ermöglicht die Integration von Lichtfelddaten in die Bild- und Videoverarbeitung. Hierzu wird die Darstellung von Lichtfeldern mit Hilfe von fortschrittlichen für 2D-Bilder entwickelten Dateiformaten erarbeitet, um die Wiederverwendbarkeit von Assets-Dateien und die Kompatibilität zwischen Anwendungen und Geräten zu erleichtern. Die neuartige 5D-Lichtfeldaufnahme und die aktuelle Forschung an Kompressions-Rahmenbedingungen werden vorgestellt. Es werden zudem verschiedene Techniken für eine optimierte Sequenzierung von Lichtfelddaten vorgeschlagen. Da Lichtfelder die vollständige 3D-Information einer Szene beinhalten, wird eine große Menge an Daten, die in hohem Maße redundant sind, erfasst. Die hier vorgeschlagenen Ansätze zur Datenvorverarbeitung erreichen dabei eine ausgezeichnete Komprimierleistung.

Acknowledgements

In the arduous journey of a PhD, you never walk alone and mine was no different. There was a sea of people who stood rock solid by me at every stage of my doctorate studies and a mere thank you is just a small word compared to the impact these people had in the successful completion of my PhD work.

First and foremost, I would like to express my deep and sincere gratitude to Prof. Dr.-Ing. Thorsten Herfet for giving me an opportunity and guiding me. His vision, immense knowledge in this field, enthusiasm and dynamism has deeply inspired me. His discipline and style of working has taught me the methodology to carry out a research work and present the result as clearly as possible. It was an honour to pursue my PhD under him.

No research work can be accomplished without the support of peers. I was fortunate to have an immensely talented group of colleagues, Christopher Haccius, Tobias Lange, Kelvin Chelli, Andreas Schmidt, and Pablo Gil Pereira. My heartfelt thanks for all the stimulating discussions and the constant support. Apart from their help, they also made work atmosphere fun. I would like to extend my thanks to two other important persons at the Lab, Zakaria Keshta and Diane Chlupka for their technical and administrative support.

Pursuing a PhD far away from family and home is challenging. The friends I have made along the journey have played a significant role by uplifting me in demanding situations and amplifying the happiness in good times. They are and will always be my second family and I could list more than a dozen but just mentioning a few close to my heart, Kaustuv Chakrabarti, Dilip Durai and Visheet Arya. Thank you everyone for being a spark in my life.

Last but the most important, the strongest pillars of my life, my Mom and Dad. Mere words cannot express my love and gratitude towards them. Everything I am today is because of them and I dedicate all my achievements to them. Thank you for the constant support through my highs and lows and for all the patience.

Contents

List of Figures

List of Tables

List of Abbreviations

5DLF	**Five** **D**imensional **L**ight **F**ield
AR	**A**ugmented **R**eality
AVC	**A**dvanced **V**ideo **C**oding
CMOS	**C**omplementary **M**etal **O**xide **S**emiconductor
DIN	**D**eutsches **I**nstitut für **N**ormung
DoF	**D**egrees of **F**reedom
DWA(A/B)	**D**ream **W**orks **A**nimation (A/B)
EPI	**E**pipolar **P**lane **I**mages
EXR	**EX**tended Dynamic **R**ange
FLIR	**F**orward **L**ooking **I**nfra**R**ed
FPS	**F**rames **P**er **S**econd
GIMP	**G**NU **I**mage **M**anupulation **P**rogram
HDR	**H**igh **D**ynamic **R**ange
HEIC	**H**igh **E**fficiency **I**mage **C**ontainer
HEVC	**H**igh **E**fficiency **V**ideo **C**oding
ILM	**I**ndustrial **L**ight and **M**agic
JND	**J**ust **N**oticeable **D**istortion
JPEG	**J**oint **P**hotographic **E**xpert **G**roup
JSON	**J**ava **S**cript **O**bject **N**otation
LCD	**L**iquid **C**rystal **D**isplay
LF	**L**ight **F**ield
LF-TSP	**L**ight **F**ield **T**ravelling **S**alesman **P**roblem
LFP	**L**ight **F**ield **P**icture
LFR	**L**ight **F**ield **R**aw
MPEG	**M**oving **P**icture **E**xpert **G**roup
MVD	**M**ulti-**V**iew + **D**epth
NIR	**N**ear **I**nfra**R**ed
PCC	**P**oint **C**loud **C**oding
PGM	**P**ortable **G**rey **M**ap
PSD	**P**hoto **S**hop **D**ocument
PSNR	**P**eak **S**ignal to **N**oise **R**atio
RGB	**R**ed **G**reen **B**lue
RGBE	**R**ed **G**reen **B**lue **E**xponent
RLE	**R**un **L**ength **E**ncoding
SAI	**S**ub **A**perture **I**mage
SLIC	**S**imple **L**inear **I**terative **C**lustering
SSIM	**S**tructural **S**imilarity **I**ndex **M**easure
UHD	**U**ltra **H**igh **D**efinition
VFX	**V**isual E**ff**ects
VM	**V**erification **M**odel

VR	**V**irtual **R**eality
VVC	**V**ersatile **V**ideo **C**oding
WCG	**W**hite **C**olor **G**amut
XML	E**x**tensible **M**arkup **L**anguage
XMP	E**x**tensible **M**etadata **P**latform
YUV	luminance(**Y**), blue-luminance(**U**), red-luminance(**V**)

Chapter 1

Preface

Immersive and interactive experience has become increasingly important in media, VFX and the film industry. According to research on physiological science, experiences make human beings happier than materialistic things [KKG14]. Creating immersive experiences enhances people's lives by fading the gap between reality and the virtual world. It attracts and engages a person into an interactive world via advanced and sophisticated technologies.

There have been several advancements in 2D video production like High Dynamic Range (HDR), Ultra High Definition (UHD) and White Color Gamut (WCG), however, they still lack life-like representations. In 3D video technology, the most popular format is the stereoscopic video composed of two views, one for each eye. Although advanced, this technology limits the user from changing the viewpoint and the depth perception, the vital element of immersive technology. On the other hand, digitally simulated immersive applications using augmented or virtual reality are well explored. However, they are still far from providing close-to-reality experiences.

The Light Field (LF) imaging technology captures the complete spatial and angular information of light rays and uses hundreds or even thousands of views with small offsets to densely sample the observation space of a scene or an object. Over the last decade, light fields for content production has gained prominence in both research and commercial environment moving immersive experience towards close-to-reality. However, the lack of state-of-the-art coding, transmission, and storage formats and also the incompatibility of the available tools to edit, post-process, and enable light field data has restricted its usage in current applications. Approaches for non-disruptive integration of light fields in image and video processing chains to facilitate its use to full potential is a core topic of interest which has been explored in this thesis.

1.1 Motivation

Since the emergence of digital photography, digital image processing and computer vision have been in the limelight. On one hand, the digital imaging devices and video acquisition hardware have evolved rapidly, but on the other hand the increasing amount of low-budget media productions have introduced new challenges to data processing approaches. Present-day influencers like to modify their captured content easily to express their creativity and artistic ideas. Therefore, with regard to the post-processing needs, it is important to conquer the constraints imposed by traditional imaging techniques and the corporeal world as such.

While traditionally only color (RGB) information is captured, the light field imaging technology allows capturing and realising light intensities passing through every point in space and in every direction, recording a huge volume of data for representation of a given scene. With the constantly advancing pixel density on devices and more newfangled processing techniques exploiting the additional information, incorporating the light field technology in end user devices is a viable option. However, the additional magnitude of flexibility attained from the light field technology comes at the expense of higher transmission capacities and storage requirements, which explicitly signifies increased data rates on all applications and devices employed to capture, display, and exchange light field content. Hence, pre-processing and compression of light field data are key elements to enable consumer usage of light fields.

In this context, the research questions addressed in this thesis work are as follows. Primarily, is interoperability possible with light field data and can it be seamlessly integrated in current workflows along with the light field post-processing capabilities intact. This requires developing techniques to adapt light field images to the existing consumer and professional file formats.

Furthermore, the need for low-complexity and feasible approaches to pre-process, compress, and store higher dimensions of light field data. Light field data contains multiple viewpoints captured within a restricted viewing angle and the perspective changes between adjacent views are consistent. Consequently, the views are highly correlated exhibiting structured redundancies both spatially and temporally. The objective is to utilise the view redundancies to achieve a better compression ratio at an optimal coding cost.

Lastly, as most of the available professional and consumer services and products are designed for conventional image content, it is essential to analyse whether the available image and video coding standards can be optimised for light field images and videos. This requires algorithms for adapting the light field data and the reference lists of the coding formats for incorporating newer imaging modalities into existing state-of-the-art standards.

1.2 Overview

This dissertation is composed of six chapters. The current chapter introduces the motivation and the problem statement that has been addressed in the upcoming chapters.

Chapter 2 discusses the basic knowledge for understanding plenoptics and light fields. The technical advances which enable light field acquisition using novel hardware and approaches are presented, including our own 5D light field camera array, as described in the conference proceeding [P9]. Further, techniques for processing the captured raw data to light fields are introduced and as well as some of the popular post-processing features of light fields are showcased.

In Chapter 3, we propose techniques to extract and transcode high resolution light field data to professional file formats like PSD and OpenEXR. The implication of the transcoding effects from the different data compression methods have been compared against the state-of-the-art compression standards and the evaluation results are presented. The work described in this chapter has been published as a conference proceeding in [P6].

Chapter 4 addresses multiple approaches for sequencing of light field data in an optimised fashion and adapting the reference lists to be compatible with state-of-the-art codecs. First, a low-complexity pre-processing solution by pseudo-temporal re-ordering of frames is presented, which maximizes the correlation between the neighbouring frames. This approach has been published as conference proceedings in [P4] and [P8]. The second technique showcases pre-processing the light field data with superpixel segmentation based adaptive Gaussian filtering and pseudo-temporal sequencing. The superpixel segmentation technique applied here is derived from our prior works [P1] and [P2] and the proposed approach is published as a conference proceeding in [P5]. Finally, an algorithm for re-ordering higher dimension light fields, including the temporal domain and generating adaptive reference lists are discussed. Using the various proposed techniques, it is demonstrated that the state-of-the-art video codecs are able to better exploit the data redundancy using both intra and inter-frame prediction. Results of this approach have been published as conference proceedings in [P3] and [P7].

In Chapter 5, the efforts towards facilitating interoperability of light field data between devices and applications at a cross-modality level is addressed. The JPEG and MPEG standardisation committees have commenced new formats JPEG Pleno and MPEG-I respectively, to support plenoptic modalities and immersive media.

Chapter 6 summarises the research findings and presents an overall view of the approaches, the algorithms developed, and the evaluations performed during the course of the studies. In addition, potential future opportunities are discussed and proposed.

Chapter 2

Towards Computational Imaging

2.1 Plenoptics and Light Fields

The concept of interpreting light rays as fields dates back to 1846, first proposed by Michael Faraday in his lecture "Thoughts on Ray Vibrations" [Far46]. Following the conceptualization, the term Light Field was envisaged by Arun Gershun in 1936. A light field is the amount of light traveling in every direction through every point in space [Ger36]. The plenoptic function parameters light rays at any given point in both space and time. Mathematically, plenoptics as defined by Adelson and Bergen [AW92], is a seven-dimensional function that describes light in terms of time (t), space (Vx, Vy, Vz), direction (θ, ϕ) and frequency (λ).

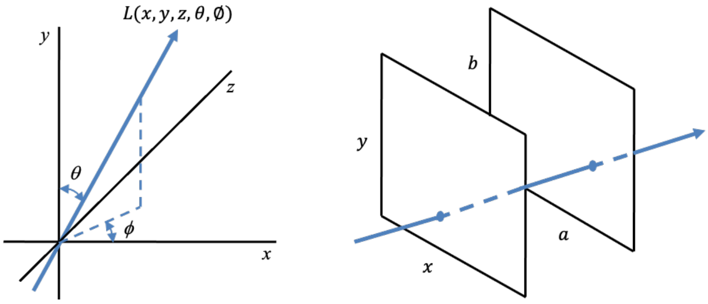


FIGURE 2.1: a) Parameterizing a ray in a three-dimensional space; b) Representing light fields in two-plane parameterization

Light field technology was introduced to the world of computer graphics in the year 1996 by Levoy and Hanrahan [LH96]. As a derivative of Adelson and Bergen's plenoptic function, rays in space are parameterized in five-dimensions. $L(x,y,z,\theta,\phi)$ where, x,y,z represent the $3D$ position in space, and θ,ϕ represent the direction, as illustrated in figure 2.1 a). Since in free space, the radiance along the light rays is constant, the plenoptic function can be deduced to a four-dimensions, representing $4D$ light fields $L(a,b,x,y)$ or photic field [MS81].

Although $4D$ light fields can be parameterized in multiple ways, the most common representation of rays is using the two-plane parameterization, as shown in figure 2.1 b). Taking the points of intersection of the light rays with an arbitrary starting point on the parallel planes, it is possible to identify the corresponding position and direction. The light rays passing in all direction can be described using this parameterization, except for the rays that are parallel to the two planes. An intuitive way of conceptualizing a two-plane light field is by imagining the ab plane as a collection of views from different perspectives, each captured from an unique observer position on the xy plane respectively.

The commonly used mathematical representation of light fields is $size(4DLF) = (s,t,u,v,c)$, where, st represents the spatial domain (horizontal respectively the vertical scene), uv represents the angular domain (horizontal receptively the vertical views) and c represents the color information. In this thesis work, we have introduced light fields of higher dimensions that include information beyond intensity and direction. To maintain consistency and to avoid ambiguity with respect to the temporal dimension, we in our works denote uv as xy for the angular domain and st as ab for the spatial domain. The time domain will be denoted with t as usual and the color information with c. The overall representation will be $size(4DLF) = (a,b,x,y,c)$ and $size(5DLF) = (t,a,b,x,y,c)$.

2.2 Light Fields Acquisition

Light fields can be captured and realised with different devices and techniques, including a handheld camera fitted with a microlens array or mirrors, an array of cameras, robotically controlled moving camera or rendered synthetically using computer graphics software. Few of the prominent approaches and devices are discussed in this section.

2.2.1 Single Sensor Cameras

Conventional cameras capture light rays as a two-dimensional and flat information. The sensor in the cameras only records the color and intensity of the light rays at each pixel position. While, plenoptic cameras record information beyond the brightness and color, including the direction of the light rays arriving at the sensor. The captured additional information can be used for reconstruction of the light rays position and aid in rendering a three-dimensional perception of the captured scene.

The principle behind a lenslet based camera was initially proposed in 1908 by Gabriel Lippmann as "Integral Photography," in which an array of small and spherical lenses with narrow baselines are used to capture a scene, to produce images from slightly different perspectives [Lip08]. The model as shown in figure 2.2 was later conceptualized as a commercial light field camera [Ng+05]. The setup consists of three major components, a main lens, a lenslet array and a photosensor. Incoming light rays from the object are focused onto the lenslet array by the main lens and the converging light rays are split by the lenslet array to form an image on the photosensor. Each

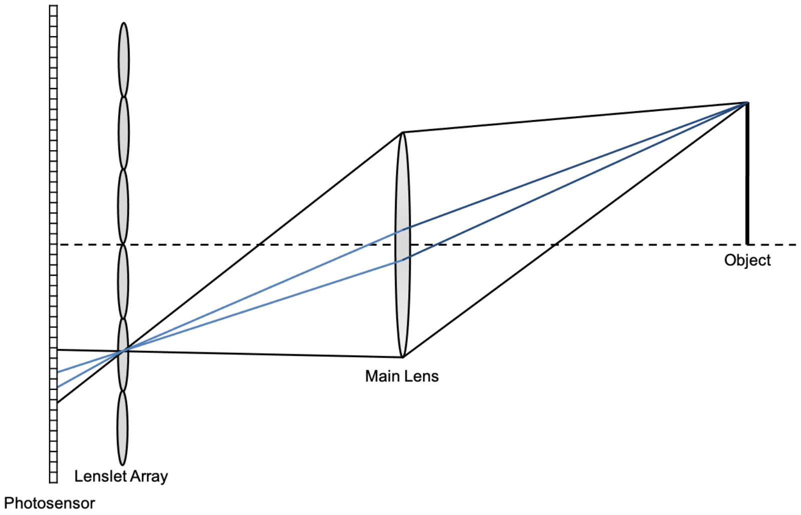


FIGURE 2.2: Lenslet camera model

lenslet forms a miniature portion of the lens aperture that measures the directional distributing of the light rays at that lenslet. The raw data as an overall view looks similar to a conventional image, however microscopically the sub-aperture images captured by each microlens are visible. This provides the direction and angular information of the light in real-world, from which the depth of the objects can be computed.

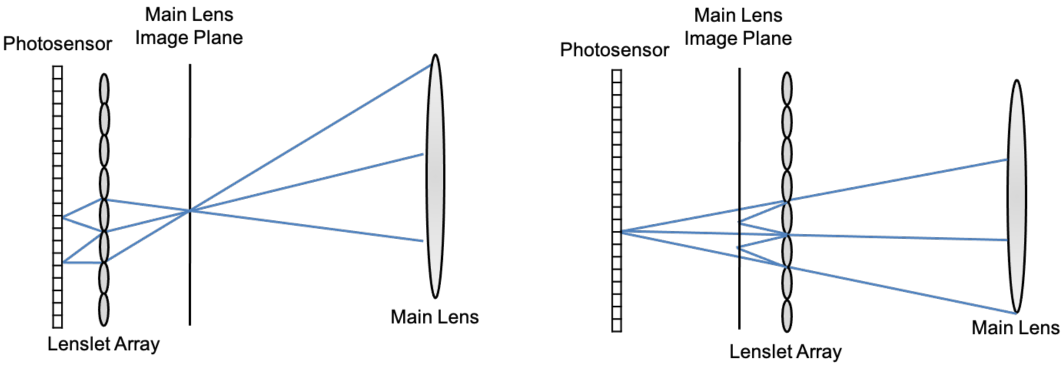


FIGURE 2.3: Two configurations of the plenoptic 2.0 cameras
a) Real image recorded in front of the lenslet array;
b) Virtual image recorded behind the lenslet array

Plenoptic cameras can be categorised into two versions based on the underlying working principle. In plenoptic 1.0 which is also the traditional plenoptic camera model as shown in figure 2.2, the lenslets are focused at optical infinity and the main lens is focused at the lenslet plane, while in plenoptic 2.0 the lenslet array is focused onto the main lens focal plane. The plenoptic 2.0 based cameras are widely known as the focused plenoptic cameras [LG09]. The focused plenoptic cameras comparatively provide a better trade-off between the angular resolution and the depth of focus, and implicitly improve the spatial resolution [PKV18]. The two different plenoptic 2.0 configurations are illustrated in figure 2.3. The primary difference is how the pixels on the sensor are ordered. Based on the placement of the lenslet array, it is possible to record a real image in front of the lenslet array, as in figure 2.3 a) or a virtual image behind the lenslet array, as in figure 2.3 b).

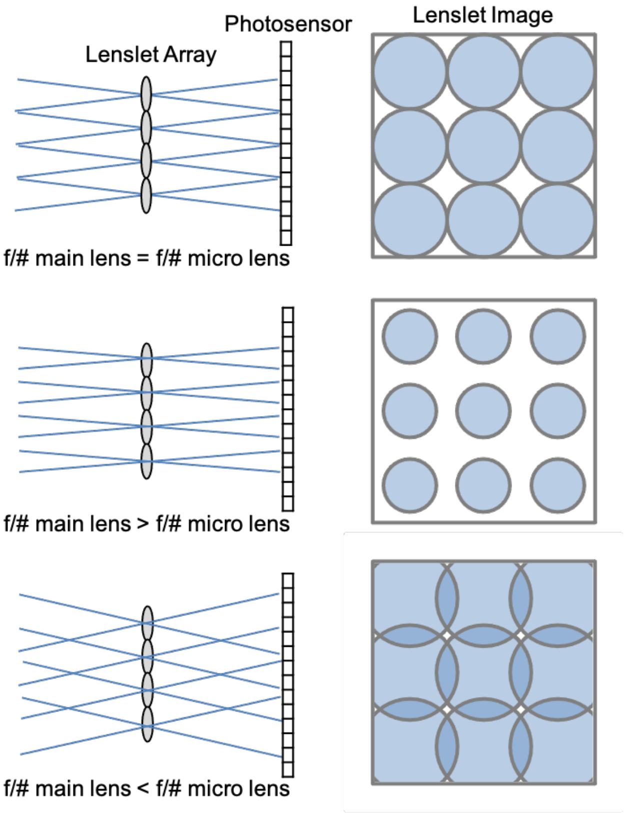

Photosensor
Lenslet Image
Lenslet Array
f/# main lens = f/# micro lens
f/# main lens > f/# micro lens
f/# main lens < f/# micro lens

FIGURE 2.4: Aperture matching between lenslet array and main lens in handheld plenoptic cameras

The significance of matching the focal ratio, f/#'s of the optics in a plenoptic 1.0 based light field camera is depicted in figure 2.4. It is important to design the relative sizes of the lenslet array and the main lens so the captured images are without reduction in angular and directional resolution. To achieve a sharp lenslet image, the lenslets are focused onto the principal plane of the main lens. As the lenslets are small in size compared to the main lens, the main lens is placed at the optical infinity and the photosensor is fixed at the focal depth of the lenslet array respectively. To maximally utilise the photosensor pixels, it is critical to choose an optimal aperture size for both the main lens and the lenslets. As described in [Ng+05], if the chosen main lens f/# is higher (smaller aperture), the recorded lenslet images are cropped, reducing the overall resolution. On the other hand, if the main lens f/# is lower (larger aperture), the resulting lenslet images overlap, producing unsatisfactory results. By matching the two f/#, the images under the lenslets are maximal in size without reduced resolution or overlapping.

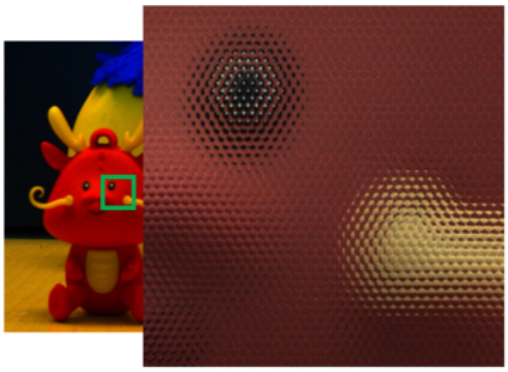


FIGURE 2.5: A sample RAW light field image captured using a handheld plenoptic camera

From figure 2.4, it can be observed that due to mismatch in packing, where spherically designed lenslets are placed in a square (rectangular) layout lenslet array, significant amount of pixels are not fully exploited. The spherical lenslets packing can be improved, for instance by approximating it in a hexagonal grid, which yields higher packing density. During processing, ray tracing approaches have to be applied for resampling the pixels to render the final images. A sample light field image captured using a handheld plenoptic camera fitted with a hexagonal grid lenslet array is shown in figure 2.5. Due to dense packing of the lenslets, the black (no information) pixels are considerably reduced.

2.2.1.1 Lytro Light Field Cameras

The Lytro is the first ever launched portable consumer light field camera. The Lytro Camera is a square and tube less device with a lens opening on one side and a LCD touch screen on the other. The different models are as shown in figure 2.6 a), majorly varying in color based on the internal storage size. The Lytro cameras are fitted with a 11 megaray CMOS Light Filed sensor. The lenslets redirects the light rays to different pixels on the CMOS sensor, which generates the angular dimension of the rays.

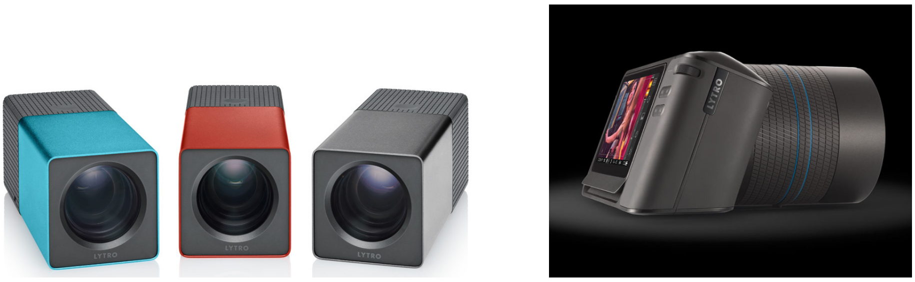


FIGURE 2.6: Lytro light field cameras
a) The Lytro camera - figure illustrated from [Lyc];
b) The Lytro Illum camera - figure illustrated from [Lyi]

Succeeding the Lytro cameras, the second generation was a professional grade light field digital camera, the Lytro Illum with a 40 megaray sensor, as shown in figure 2.6 b). The Lytro Illum Cameras have a 8x zoom lens which is similar to a 30-350mm digital camera lens and the camera can focus from 0mm to infinity. An important add-on was the display, featuring a live view overlay of the objects at multiple depths and their refocusable range. Both the Lytro cameras are based on plenoptic 1.0.

Lytro further introduced light field solutions for virtual reality, the Lytro Immerge and for cinematic content, the Lytro Cinema, a first of its kind. Although with these products, Lytro pioneered a progressive transformation from traditional 2D capturing to 3D volumetric video, they have been taken over by Google and have suspended the productions, while we continue to research with the data generated using these novel devices.

2.2.1.2 Raytrix

Raytrix manufactures light field cameras for scientific and industrial applications. The Raytrix cameras simultaneously record the 2D information of a scene, along with the metrically calibrated depth information. The processing is performed using their proprietary software to digitally manipulate the captured information. While the underlying conceptual model of the different Raytrix cameras are the same, as shown in figure 2.7, there are several additional adaptable features such as extending up to 65 MegaRays at 71 FPS, different sensor sizes, resolutions, frame rates, and also versions that capture Near-Infrared and mono information. Several of the latest Raytrix cameras are conceptualised based on plenoptic 2.0.

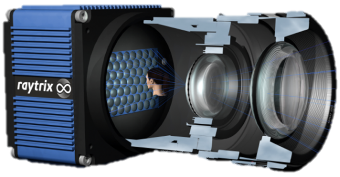


FIGURE 2.7: Raytrix 3D camera
Figure illustrated from [Ray]

2.2.2 Camera Arrays

The fundamental geometry of a multi-camera array setup is illustrated in figure 2.8. Only the vertical dimension is illustrated for simplicity. As described in our research work [P9], to achieve a desired number of views and exploit the captured data as light fields, the frustrum of the individual cameras has to overlap. Considering N as the number of cameras and with the given multi-camera array geometry N^2 views can be captured by recording rays from N^2 directions. Occlusion of the rays can occur based on the convex hull of the capturing scene and the effective f-number of the individual cameras is only a $\frac{1}{N}$ of the overall focal ratio, widening the resulting depth of field.
Light fields captured using a multi-camera array are sparse compared to a hand-held plenoptic cameras that record dense light fields. The relatively

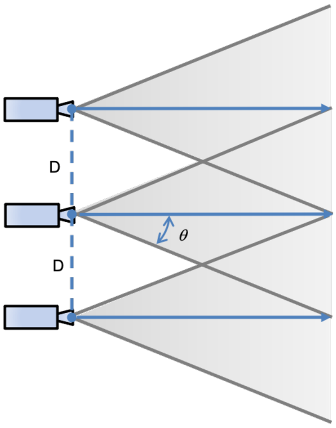


FIGURE 2.8: Multi-camera array geometry

high packing density of the microlens array densely samples the acquired light field. As the sensors in a multi-camera array system cannot significantly overlap due to the large physical baselines between the cameras depending on its grid layout, the acquired light fields are sparsely sampled. Conceptually, the number of microlens in a plenoptic camera is equivalent to the resolution of a single view from a camera array, and similarly, the number of pixels behind each lenslet in a plenoptic camera is equivalent to the number of views in a camera array.

2.2.2.1 The Stanford Multi-Camera Array

A reconfigurable multi-camera array system was first introduced by Stanford University [Wil+05]. The goal of the setup was to overcome the restrictions of conventional cameras and build a cost efficient virtual camera setup that represents a computational imaging system. The Stanford multi-camera array depicted in figure 2.9, consists of a 100 CMOS sensor based cameras that capture and deliver content synchronously and in real-time.

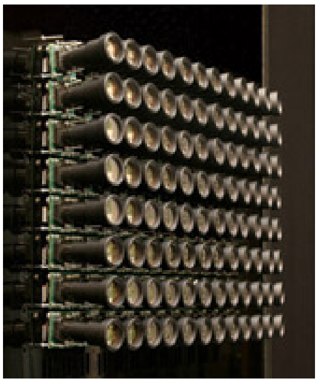


FIGURE 2.9: The Stanford multi-camera array
Figure illustrated from [Wil+05]

As discussed in [Wil+05], a multi-camera array system can vary in functionality based on the physical configuration of the cameras. By placing the individual cameras within a close proximity enables the system to operate as

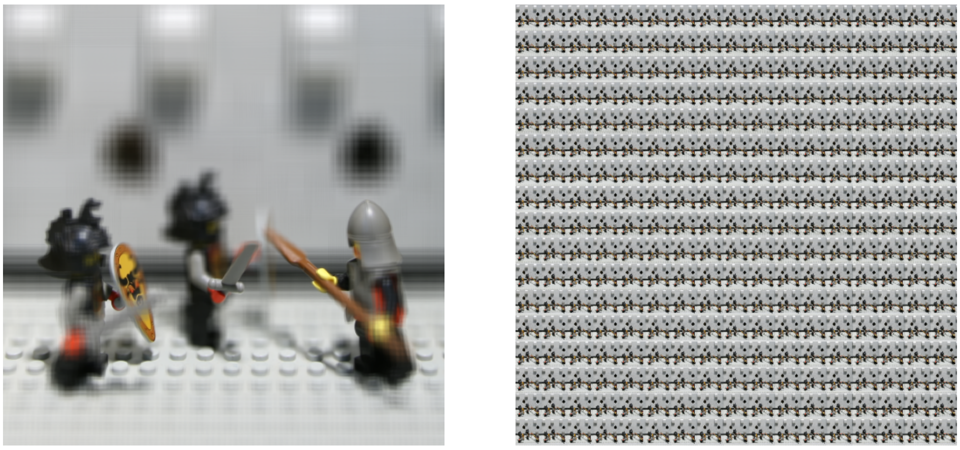


FIGURE 2.10: Lego knights from the Stanford light field archive
Dataset source [VA08a]
a) Lenslet view; b) Sub-aperture images (SAIs)

a single-center-of-projection synthetic camera, while if the cameras are configured over a wider proximity the system operates as a multiple-center-of-projection camera, capturing light field content. Figure 2.10 shows a multi-camera array asset from the (New) Stanford light field archive. The left image, figure 2.10 a) shows the lenslet view (each pixel is an assembly of 17x17 rays) and the right image, figure 2.10 b) shows the combined sub-apertures (17x17 images or 1024x1024 pixel).

2.2.2.2 The 5D Light Field Camera Array

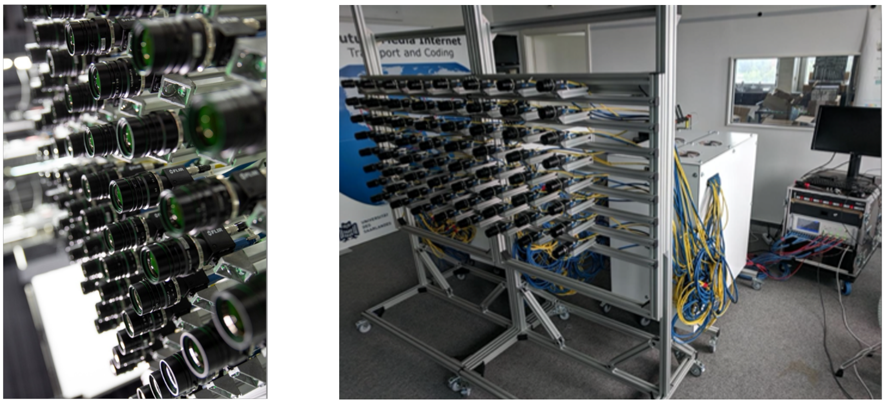


FIGURE 2.11: The 5D light field camera array

Our own 5D light field camera array (conceptualised and assembled at the Telecommunication Lab, Saarland University, Germany) is shown in figure 2.11. The array consists of 64 FLIR Blackfly cameras, capturing light field videos at 40fps with a resolution of 1920 x 1200. The customised rig allows the cameras to be reconfigured to different layouts within a specified range as described in table 2.1, introduced in our research work [P9]. With custom electronics the exposure/triggering time of the individual cameras are controlled independently, allowing the camera array to capture 5D information, i.e. 4D rays plus the temporal resolution.

TABLE 2.1: The 5D light field array parameters

Parameters	
Sensor Type IMX249	
Sensor size (diagonal)	$13.4mm$
Pixel width	$5.86\mu m$
Resolution (horizontal)	1920 pixel
Resolution (vertical)	1200 pixel
Aspect Ratio	1.592
Framerate	$40fps$
Lens	
Focal length	$12.5mm$
Aperture	$f/1.4mm$
Shutter Control	
Min. exposure time	$19\mu s$
Control of exposure time	$10\mu s$
Array dimension	
Number of cameras	64
Spacing (horizontal)	90–550mm
Spacing (vertical)	90–250mm

Other notable multi-camera array systems are, a self-reconfigurable camera array [ZC04], the PICam [Ven+13] and the ProFUSION-25C3 camera array [Hua+15].

2.2.3 Other Techniques

Apart from the above discussed two widely popular categories, there are few other devices and techniques that can capture light fields.

2.2.3.1 Camera gantries

In camera gantries, the camera is mounted and moved using a supporting structure and captures the scene at regular intervals. The displacement between the shots and the camera resolution correspond to the angular and spatial sampling. While, the light fields captured by gantries are sampled similar to camera arrays, the cost and overall effort are reduced considerably. Nevertheless, such gantries are constricted to only capture static light fields. Two popular gantries from the Stanford Laboratory are the Light Field Gantry, shown in figure 2.12 a) and the Lego Mindstroms Gantry, shown in figure 2.12 b) respectively.

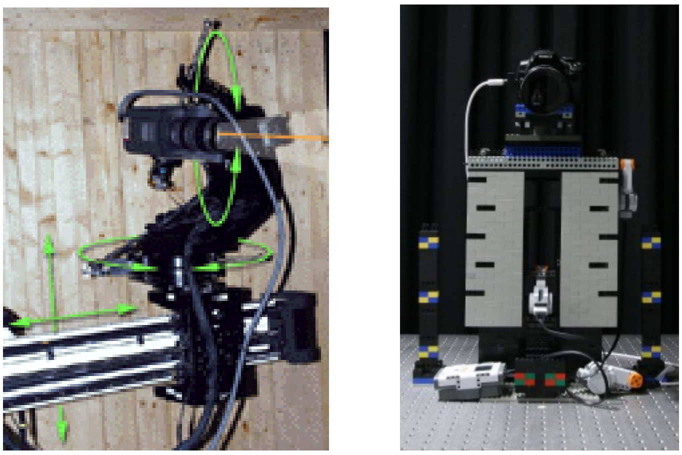


FIGURE 2.12: Camera gantries
a) Light field gantry; b) Lego mindstorms gantry
Figures illustrated from [VA08b]

2.2.3.2 Mirror based systems

Instead of a microlens array, some devices use a mirror system to capture light fields. The K|Lens technology [Kle] uses mirrors in a tunnel setup (like a kaleidoscope) to separate the light beams to form multiple perspectives, achieving the best of both microlens technology and multi-camera array. K|Lens commercialises the technology as a camera add-on product that can be used with any standard digital camera turning the device to capture light fields, as illustrated in figure 2.13. Few other devices that use mirrors rather than a microlens array to record light field are described in [Tag+10] and [Tsa+17].

FIGURE 2.13: K|Lens camera add-on
Figure illustrated from [Kle]

Finally, a unique device that combines both the camera gantry and the multi-camera array technology is proposed by Google, generating panoramic light fields [Ove+18] for AR and VR content.

2.3 Processing of Light Fields

The captured raw sensor data using both the lenslet based cameras and the multi-camera arrays undergoes several processing steps to be realised as light fields. The *ab* and *xy* planes vary with respect to the light fields acquisition method. For single sensor based light field cameras, *ab* denotes the lenslet plane and xy denotes the main lens. On the other hand, for camera arrays capturing light field video, ab represents the individual camera lens and xy represents the scene plane, while the additional time component is defined by t. An intuitive technique for rendering the data obtained using the plenoptic cameras is by decoding the raw 2D microlens images to a 4D light field format from which 2D slices (views) can be extracted. In accordance to the two plane parameterization discussed in section 2.1, the figure 2.14 illustrates slicing of a light field image to form multiple sub-aperture views, as initially presented in [LH96]. Here, consider the xy plane as a set of pinholes or lenslets and the ab plane as the image sensor or pixel plane. By fixing a constant x & y and considering all the a & b values, renders a sub-aperture view of the scene as captured by a pinhole at the position (x,y), as illustrated in figure 2.14 a). While, fixing a & b and considering all possible values for x & y shows every pixel at position (a,b) from the different views, as illustrated in figure 2.14 b). Another interesting way to visualise light fields is using the Epipolar Plane Images (EPIs), initially proposed in [BBM87] for spatio-temporal volume analysis. The epipolar plane images depict the intensities of the pixels in terms of vertical or horizontal angular and spatial coordinates.

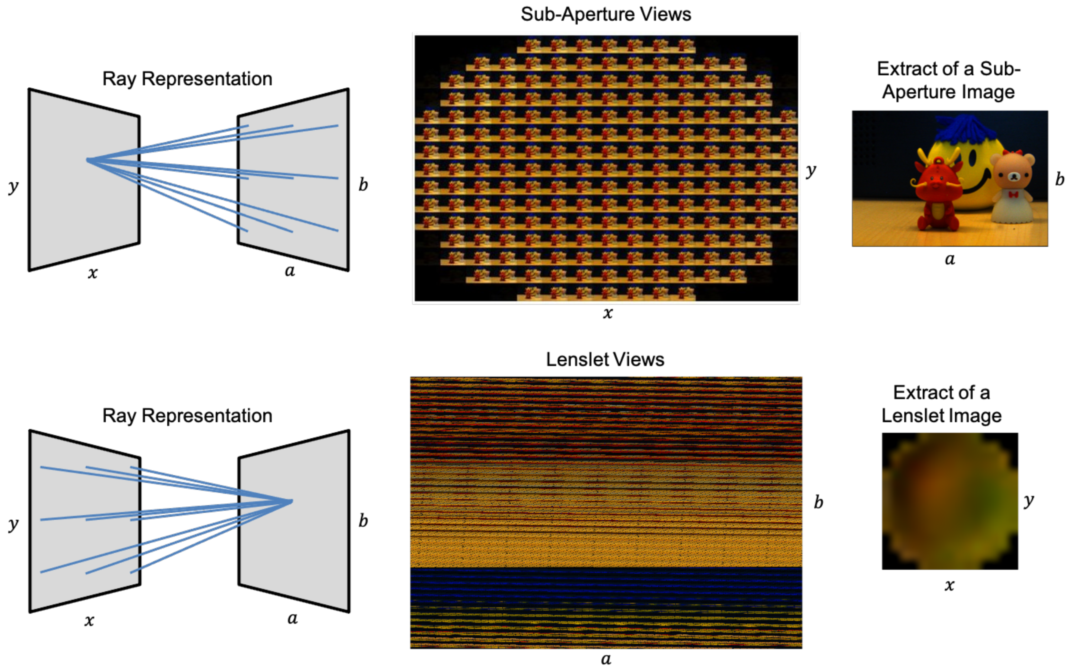


FIGURE 2.14: 4D light fields visualisation
a) Sub-aperture views; b) Lenslet views

Processing lenslet-based plenoptic camera light fields have been researched in detail by Dansereau et al. [DPW13] and their light field toolbox for MATLAB [Dan] provides a bunch of tools to decode, calibrate, rectify and manipulate light fields. In their works, the main lens of the lenslet-based plenoptic camera is modelled as a thin lens and the microlens as an array of pinholes. As the grid placement of the lenslet array varies from an image to another based on the camera settings, a set of white images (images captured through a diffuser) with different focus and zoom settings are captured prior. The white images, due to vignetting has a bright spot in each lenslet, approximating it as the pinhole centers. These lenslet centers are used for estimating the grid parameters such as the vertical and horizontal spacing, offsets and shifts, which is applied during decoding to compensate the non-integer microlens spacing and the translational and rotational offsets.

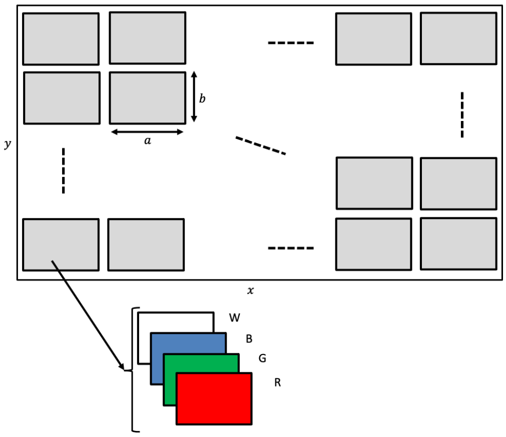


FIGURE 2.15: 4D light field data structure $[x * y * a * b * 4]$

The initial step for decoding a lenslet-based light field to a 4D unrectified light field is demosaicing the raw lenslet image to reconstruct a full color image and then divide it by an appropriate white image (chosen based on closely matched camera settings) to correct for vignetting. Using the determined grid parameters, the lenslets are resampled, overcoming the non-integer spacing, scaling and rotational imbalances. This aligns all the microlens centers to the pixel centers. Then the individual lenslets are sliced into equally sized rectangles which slightly overlap with each other due to the close packaging density of lenslets . Now the raw 2D lenslet image is in a 4D format and has a hexagonal sampling at lenslet indices a,b and rectangular pixels at pixel indices x,y. The next steps primarily focuses on interpolation. The lenslets are interpolated from a hexagonal grid to a rectangular grid (this step can be ignored if the lenslet array has a square or rectangular packaging), which also compensates for the horizontal and vertical sampling offsets. Then the interpolation is performed at the pixel indices to correct for the non-square pixels. Finally, the outlier pixels are masked off, resulting in an unrectified 4D light field. Figure 2.15 depicts the 4D light field structure consisting of 2D RGB images. A weighting image W, containing a confidence score for each pixel, is also included in the structure. The horizontal and the vertical resolution of the sub-apertures are represented by $a * b$, whereas the number of sub-apertures in both directions are depicted by $x * y$. The factor 4 represents the channels RGB plus the weighting image W. The W channel is used in filtering applications that accept a weighting parameter. For example, while estimating histogram equalization for adjusting the brightness of a light field images, the W channel can be used for ignoring the zero-weight pixels.

Calibration and rectification of the decoded light fields is very essential to overcome the lens distortions and allow accurate feature matching between sub-aperture images. For simplicity, in this thesis work, the rectified light fields are mostly used. Some of the prominent research works on plenoptic camera calibration include, generating a 5 ∗ 5 homogeneous intrinsic matrix that relates each spatial ray to its respective pixel indices [DPW13]. An approach for calibrating focused plenoptic cameras by metric analysing the captured scene [Joh+13]. A method that estimates the orientation and position of the lenslet array using a calibration image and camera geometry [TFT14]. Another recent work proposes a hybrid calibration model considering both the microlens array and the main lens geometry [DBM19].

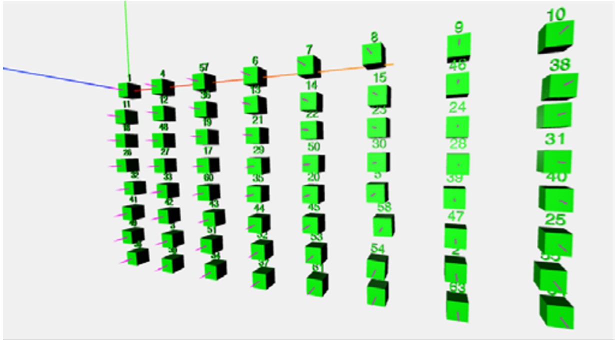


FIGURE 2.16: 3D representation of camera coordinates
Figure illustrated from [P9]

On the contrary, in the case of multi-camera array systems, the 4D representation of light fields is straightforward and does not require a multi-step decoding process, but the challenges are due to optical aberrations, misalignment and irregular spatial and angular sampling. The cameras have to be calibrated very precisely, as even the smallest variation in the coordinates results in angular correspondence errors between views. This causes superposition of rays from wrong direction, producing severe visible artefacts and as well as poor geometrical reconstruction [Via+17]. For multi-camera arrays, every individual camera requires a complete camera matrix with all intrinsic and extrinsic parameters, as they are device dependent. Figure 2.16 illustrates the 3D representation of our 5D light field camera array's camera coordinates, as introduced in [P9]. For initial experimentation, an OpenVC implementation based on the work [Zha00] was used to estimate the camera parameters. It can be observed that, in X and Y directions the geometrical transformations are precise for many cameras, while there are large deviations in the Z direction. The errors majorly occur due to inaccurate feature detection arising from changes in camera focus and the surround lighting. Several iterations of refinement have to be performed based system specific issues to obtain accurate results.

Further, the most common processing steps for the data captured using a multi-camera array includes, demosaicing the raw sensor data to retrieve the full color range. This process is applied on all the views from the individual cameras. Then, color alignment is carried out to eliminate the different camera responses. To overcome the camera alignment errors, the views are accurately rectified using the parameters from calibration. Finally, the rectified views are stored as uncompressed light field data.

As we have looked into the challenges and the need for a tedious processing pipeline to convert raw sensor data to light fields, rendering them photorealistically is another option. By synthetically rendering, we can precisely program the geometry, texture, viewpoints and lighting of the virtual scene. A major effort is only required in creating sophisticated 3D models for the scene file. Generating ground truth and depth/disparity maps are as well trivial and produce accurate results.

2.4 Features and Applications

Capturing light fields over 2D images, opens up several new avenues on how the data can be creatively utilised. Even the basic features of an image can be changed after capturing. While, light field features extends manifold, some of the core ingredients are discussed briefly.

Digital Refocus: The ability to refocus after the picture has been captured is one of the remarkable feature. By recording and algorithmic manipulation of the angular dimension of the rays (combine rays at the same point from different lenslets or camera views), it is possible to reconstruct multiple versions of a picture with adjustable focus. On the other hand, there exists a refocusable range limitation over every image. The refocusable range consists of all the rays focused relatively sharp after the image has been captured. Figure 2.17 illustrates how the light field image, shown in figure 2.5 can be refocused at different depths.

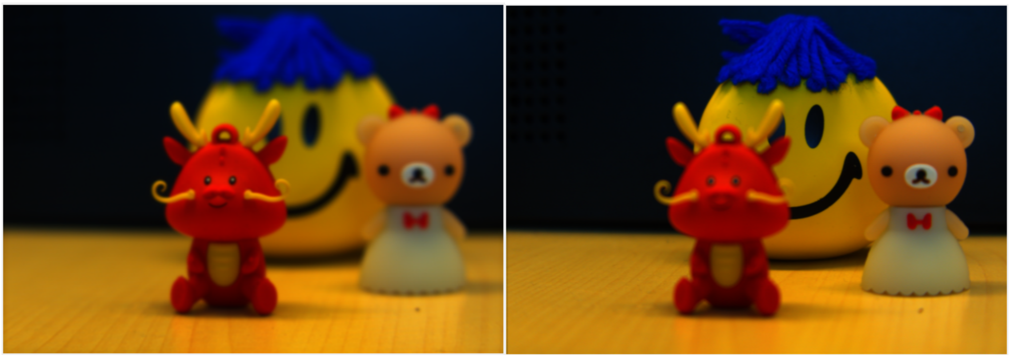


FIGURE 2.17: Digital refocusing
a) Refocused on foreground; b) Refocused on background

Varying the Depth of Field: Another interesting concept is the synthetic aperture. By capturing 4D light fields we implicitly have a 3D model and the direction of the rays. Using this knowledge, the optical parameters can be modified computationally, reproducing the angular integration among the rays which are fixed in conventional cameras. The aperture plays a key role in determining the depth of field in an image and can be synthetically adjusted to render scenes with infinite (all in focus), shallow (blurry or bokeh effect) or at other desired depth of field.

Moving the observer: Resampling particular light rays that travels through different microlens or camera views, it is possible to generate multiple perspectives of the captured scene. Figure 2.18 shows the light field picture 2.5 from two different perspectives (the highlighted blue regions clearly showcases the shift). The perspective shift effect is similar to viewing the scene from varied lines of sight and gives the user an immersive experience. Considering a movie theatre scenario, this parallax effect can give the viewers the possibility to slightly move their head seated at their respective places and get a sneak peak of the hidden protagonist or an object.

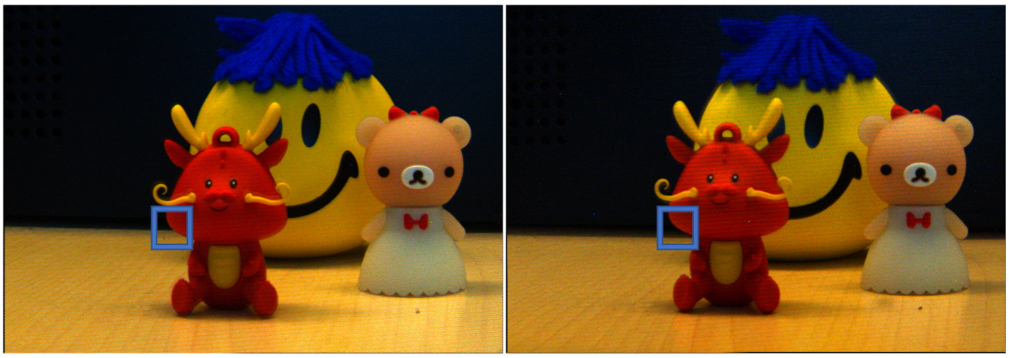


FIGURE 2.18: Perspective shifts - two different sub-aperture images from the same captured scene

Depth sensing: In general 3D depth camera systems, stereo vision, structured light, or time-of-flight cameras are used for depth sensing and requires expensive processing. Instead of specialised equipment, from light field data the relative depth of the objects in the scene can be used to precisely produce the distance imaging.

Few other interesting applications with light fields are, recreating 3D scenes, depth scaling, enhanced segmentation, reducing the glare in images from the lens optics, vision based robot control, material classification and also the use of light field microscopy in the medical field like analysing neural activity. All the mentioned features are possible with both static (still images) and dynamic (videos) light fields. The biggest challenge is on the processing end, requiring high computing energy, sophisticated software and efficient compression, storage and transmission techniques, to bring light fields fully to the consumers.

Chapter 3

Representation of Light Fields in Existing Formats

Unlike conventional 2D images which only record the color information, light fields record both the color intensity and angular information of the scene and therefore it is essential to adapt the light field data to be handled by file formats designed for 2D image assemblies. Adapting the light field data to the current and widely used imaging file formats like PSD and OpenEXR, allows asset re-usability and also interoperability between devices and imaging applications. In this chapter, the techniques to extract and transcode high resolution light field data to professional file formats are investigated and the implication of the transcoding effects are evaluated by comparing different data compression methods offered by these formats in contrast to the state-of-the-art compression standards.

3.1 The Terminology: 4D, 4.5D and 5D Light Fields

4D light fields: A light field with four dimensions is an assembly of views recorded from multiple perspectives. The most intuitive way of understanding is with the intersection of a ray with two defined planes, introduced in section 2.1, describing the two angular and two spatial coordinates respectively. Conceptually, the difference between light fields and multi-view data is the requirement of the camera extrinsic parameters. Using the metadata, i.e., the viewing direction and position of the camera, the 4D coordinates are mapped 1:1 to the captured rays respectively (taking into account the direction and the position of the rays). Figure 3.1 shows a 4D light field image captured using the Lytro Illum camera and processed with the MATLAB light field toolbox. The corrected lenslet view where the hexagonal structure of the lenslets is visible is shown in figure 3.1 a) and the assembly of 2D sub-aperture images rendered from the corrected lenslet view is shown in figure 3.1 b).

4.5D light fields: In the case where all cameras are gen-locked, i.e., synchronised at a matching temporal sampling frequency and sampling phase, 4.5D light fields are captured. This is merely recording a light field video. Figure 3.2 illustrates two different frames from the Unfolding dataset recorded using the 5DLF camera array (introduced in section 2.2.2.2). Since all the

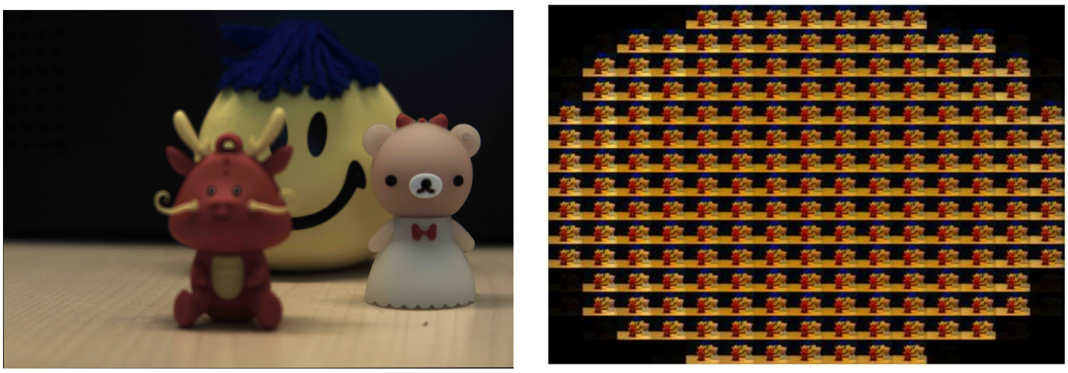


FIGURE 3.1: 4D light fields captured using the Lytro Illum and processed with the Matlab light field toolbox
a) Corrected lenslet view; b) Sub-aperture views

camera phases are constant, it can observed there are no temporal change between the individual camera views but only perspective changes.

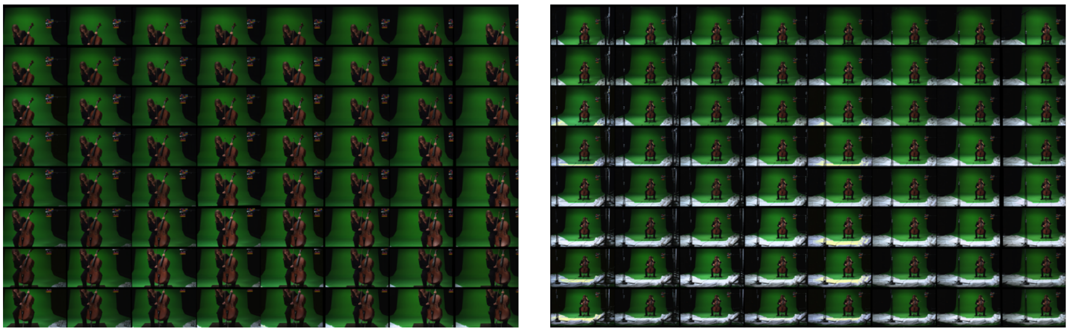


FIGURE 3.2: 4.5D light fields captured using the 5DLF camera array. Unfolding 2.0 dataset - a) Frame 200; b) Frame 300
Dataset source [Unf]

5D light fields: Considering scenarios where the scene to be captured is not static but has objects that are dynamic, recording the temporal information becomes critical. While 4.5D light fields are a subset of 5D light fields, the rays or the assemblies of rays when captured at varied time instances (programming the individual cameras to trigger at specified time intervals) records the 5^{th} dimension. The scene can be captured with different spatio-temporal configurations.

To simulate 5D light fields with a still image camera (Lytro Illum), we created so called stop-motion movies. A scene with a predefined motion per frame is created and the original full light fields are sub-sampled by the pattern. An example of four motion phases and each phase sub-sampled by a factor of four is shown in figure 3.3 a) and b) respectively and we have introduced it in our research work [P9]. To record novel 5D light fields, we have created the scene HaToy, shown in figure 3.4 a). The scene is built with numerous static and moving objects of variable size and geometrical complexity and

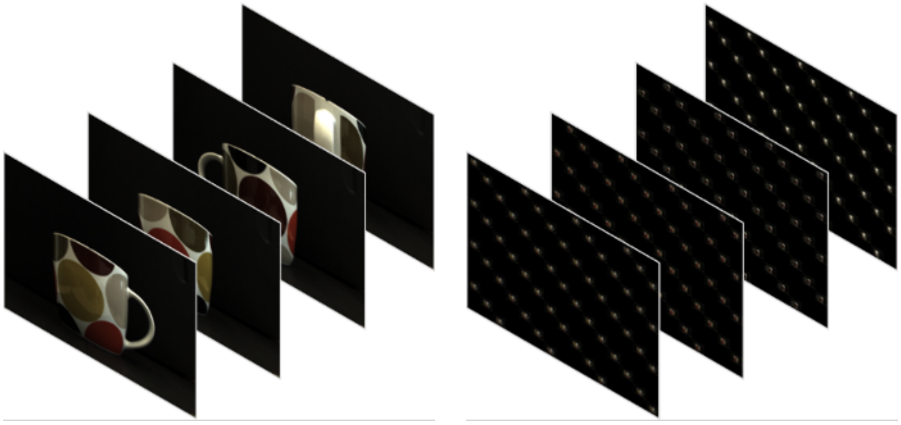


FIGURE 3.3: Stop-motion light field asset
a) Phases; b) View Assemblies

is captured with different spatio-temporal configurations, as shown in configurations, as shown in figure 3.4 b). A closer look on the motion phases and more intricate details about the dataset and the capturing configurations [HLC19] will be discussed in chapter 4.

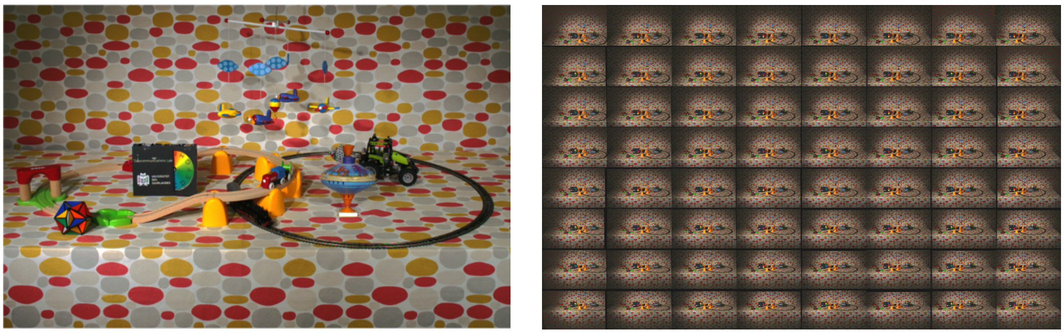
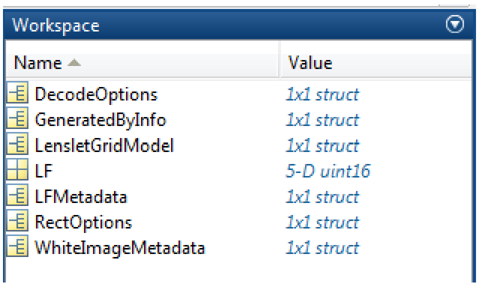


FIGURE 3.4: 5D light fields captured using the 5DLF camera array; a) A camera view; b) Sub-frame views

3.2 Light Field Formats

As pioneers in handheld plenoptic cameras, Lytro developed novel file formats for processing and storing light fields. Light Field Raw (LFR) format is a Light Field Picture (LFP) containing the raw image data that has to be processed to decode the light fields. The RAW LFR contains both the sensor data i.e., the pixel values and the frame metadata. The frame metadata is very crucial for the reconstruction of the image and to maintain the underlying properties of light fields while representing light fields as 2D assembly of images. It includes essential information such as the calibration parameters, sensor readings, hardware configurations, frame parameters. A sample of a decoded light field structure using the LF toolbox is shown in figure 3.5, containing both frame data and metadata. On the other hand, with multi-camera array systems the raw sensor data is encoded and stored in the available state-of-the-art lossless image file formats and the accompanying metadata (camera matrices) are coupled in external files in machine readable formats.



Name	Value
DecodeOptions	1x1 struct
GeneratedByInfo	1x1 struct
LensletGridModel	1x1 struct
LF	5-D uint16
LFMetadata	1x1 struct
RectOptions	1x1 struct
WhiteImageMetadata	1x1 struct

FIGURE 3.5: Sample of a decoded light field structure containing both image data and metadata

3.3 PSD Format

A Photoshop Document (PSD) is an advanced imaging file format native to Adobe Photoshop. The PSD format is popular for professional graphics designing and storing high quality data. It supports multiple image layers, layer masks, file information, metadata, keywords, annotations, adjustment layers and other imaging options. A PSD file can contain a maximum width and height of 30.000 pixels and can store files upto 2 gigabytes in size. Using the PSD format is ideal for creating 2D image layers, work on them individually and export them to other image file formats for distribution. The file structure of the PSD format is illustrated in figure 3.6. The format consists of five primary sections, file header, color mode data, image resources, layer and mask information and, the image data. The file header has a fixed length and stores the fundamental properties of the image. The color mode data contains information whether the image data is color or dual-tone. Image resources block stores the non-pixel information. The metadata associated with the images are saved in the image resources section in Extensible Metadata Platform (XMP) format, which is built on Extensible Markup Language (XML). The XMP format facilitates the use of metadata over cross-platform applications. The fourth section of the format contains information about the image layers, mask parameters, channels in the layers and the associated values. The pixel data is stored in the final image data section in scan-line order, arranged in respective color channels. More detailed description on the specifications can be referred to [Kno19].

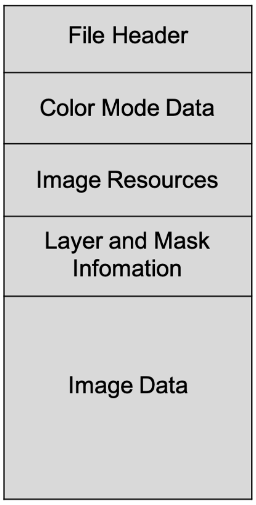


FIGURE 3.6: Photoshop document file structure

3.4 OpenEXR Format

The OpenEXR format developed by the Industrial Light and Magic (ILM) is an open source standard. It is a powerful file format used widely in several computer imaging applications and the VFX industry. The format is designed to support High Dynamic Range (HDR) imaging, multi-channel raster and allows multiple pixel sizes such as 32-bit unsigned integer, 16-bit and 32-bit floating point values. OpenEXR offers significant advantages over conventional multiplexed images and video files in the media environment and special effects. With the ability to store multiple layers, the standard ensures maximum flexibility in high end 3D compositing programs. Multiple channels of an image or 2D image assemblies can be stored as one entity which makes the OpenEXR format speed efficient, as it is feasible to cram as much information as required within one file. The standard also provides several data compression options in both lossy and lossless formats. The file structure consists of two primary sections, file header and the image data as shown in figure 3.7. In OpenEXR format, an arbitrary number of additional attributes can be stored along with the pixel data, which facilitates the consolidation of the accompanying metadata within a single file. The document on OpenEXR file layout [Kai13], offers extensive information on the supported data types and the file attributes.

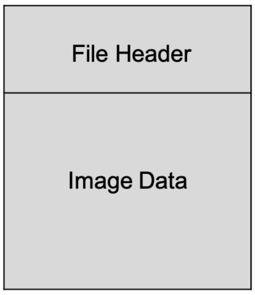


FIGURE 3.7: OpenEXR file structure

3.5 Transcoding Procedure

Transcoding is the process of digital-to-digital format conversion. The procedure is very frequently used in media and broadcasting sectors to increase the compatibility of the data with multiple different target devices and workflows. Transcoding supports in overcoming the limitations over transmission, storage capacity, adapting to novel formats, asset re-usability and provides cost efficiency. As light fields are contemporary data and the commonly used formats and production engines are made to be compatible with conventional data, it is very essential to transcode light field data to be suitable and deployed using the available professional and consumer applications. In this section, our implementation for transcoding the light field data to OpenEXR and PSD file formats, which are extensively used in several current production workflows are discussed.

Light field data captured using both plenoptic handheld cameras and multi-camera arrays are dealt with in our works. The captured light fields are first analysed, to understand the fundamental data structure and to perform transcoding without losing the underlying properties of light fields. Our file format converts are programmed in Matlab and are very robust in terms of handling huge datasets and performing the transcoding time efficiently. The pipeline is fully automated, requiring only the file location of the image data and the accompanying metadata. The file format conversions are programmed with backward compatibility, i.e., our algorithms can both read and write light field data to PSD and OpenEXR formats and vice versa seamlessly.

Matlab provides no implicit support to read or write PSD and OpenEXR file formats. Both the standards have been extensively studied and programmed to fully functional codes that can read and write data in Photoshop document and OpenEXR file structure respectively. The core steps for transcoding to advanced imaging file formats and inverse are illustrated in figure 3.8. The block diagram represents an overview of the implementation and showcases the common procedures performed in both PSD and OpenEXR conversions. Light fields recorded using plenoptic cameras are decoded to 4D light field structure, example shown in figure 3.5 (2D assembly of images + the metadata). The multi-camera array recorded light fields are structured in separate sub-folders (respective cameras + associated metadata and the number of recorded frames). Once the light field data are decoded, the process begins with extracting the essential non-pixel information such as resolution, frame count, color mode and mask parameters. This information is parsed sequentially to the different sections and the process runs until all the image data are fed in. The intermediate steps corresponding to the proposed two file format conversions are discussed exclusively in the below sections.

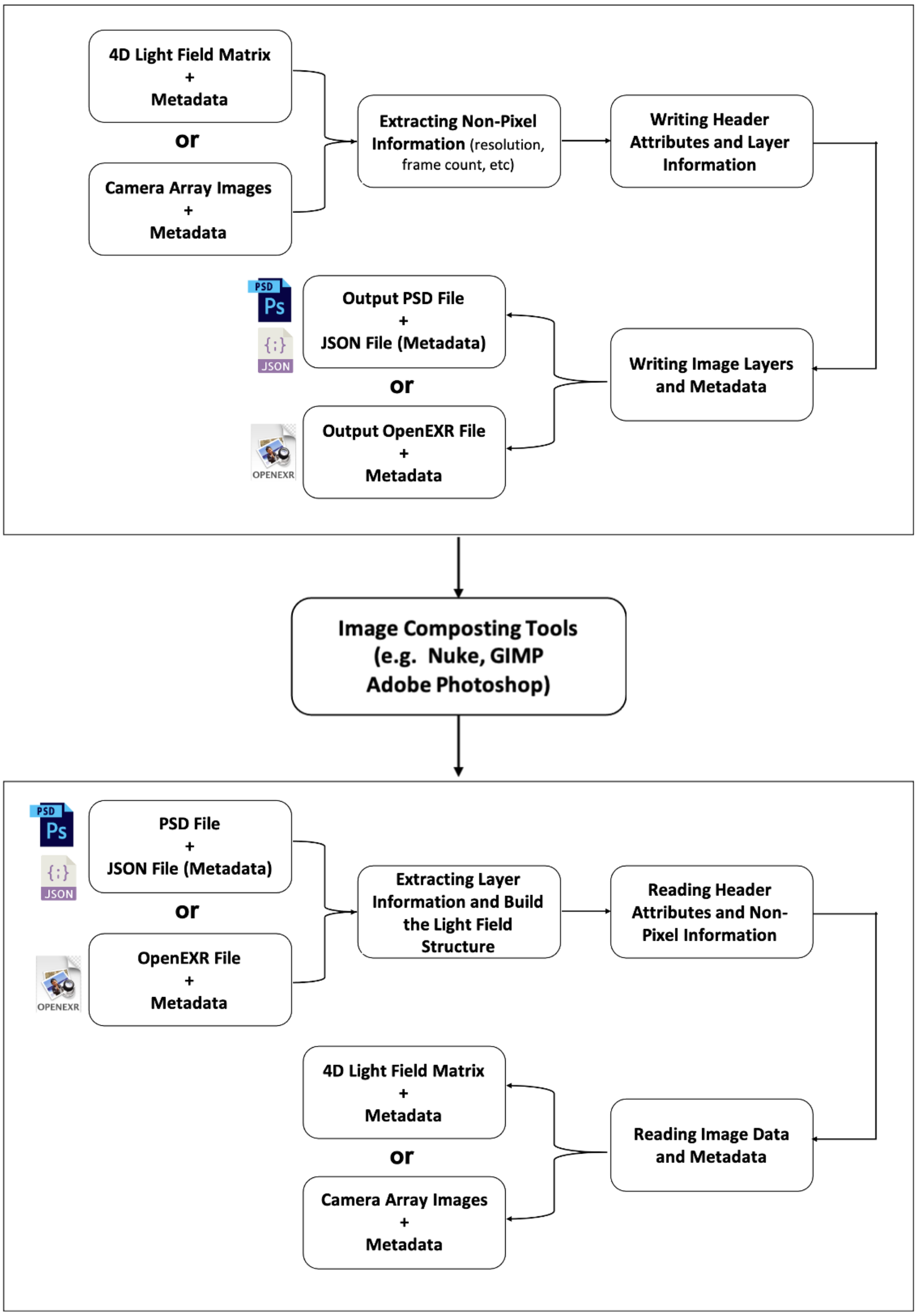
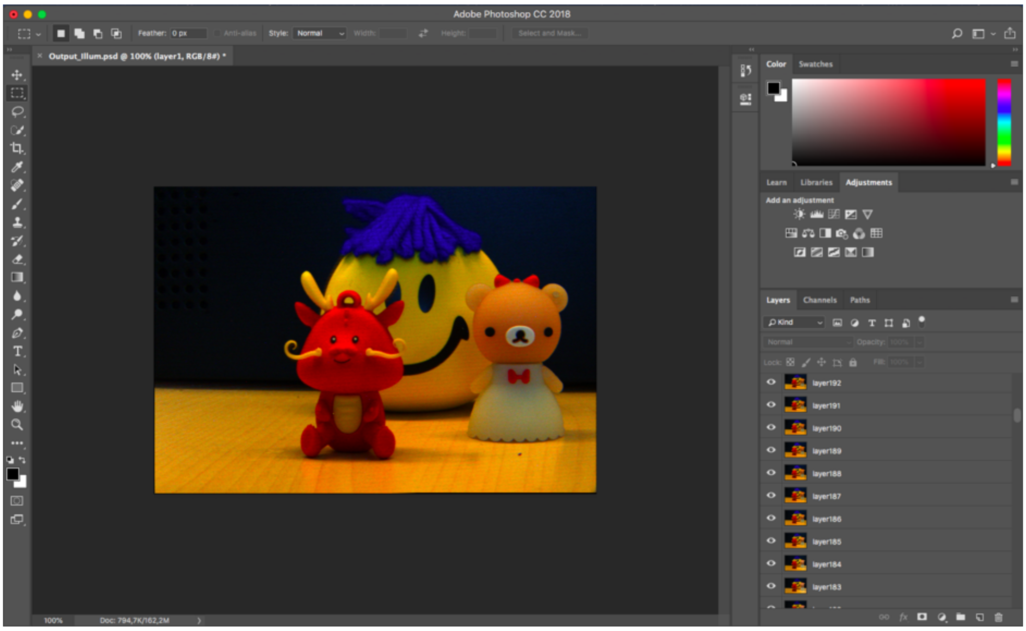

4D Light Field Matrix + Metadata
or
Camera Array Images + Metadata
Extracting Non-Pixel Information (resolution, frame count, etc)
Writing Header Attributes and Layer Information
PSD
Ps
{;}
JSON
Output PSD File + JSON File (Metadata)
or
OPENEXR
Output OpenEXR File + Metadata
Writing Image Layers and Metadata
Image Composting Tools (e.g. Nuke, GIMP Adobe Photoshop)
PSD File + JSON File (Metadata)
or
OpenEXR File + Metadata
Extracting Layer Information and Build the Light Field Structure
Reading Header Attributes and Non-Pixel Information
4D Light Field Matrix + Metadata
or
Camera Array Images + Metadata
Reading Image Data and Metadata

FIGURE 3.8: Transcoding procedure

3.5.1 Light Fields <–> PSD

As PSD file format supports image layering, the sub-aperture images or the camera views are written as PSD layers, encapsulating the entire image data as a single PSD file. Although PSD format supports image metadata, it comes with default template fields which are incompatible with most of the light field metadata fields. Hence, the associated metadata is written and exported as a JSON file coupled with the corresponding PSD file. The transcoded light field data is now suitable to be exploited by popular imaging tools such as

GIMP, Nuke and Adobe Photoshop. A light field transcoded to PSD file format and accessed using Adobe Photoshop is illustrated in figure 3.9. The different sub-aperture images can be viewed as individual layers in the side tab.



Feather: 0 px

Style: Normal

100%

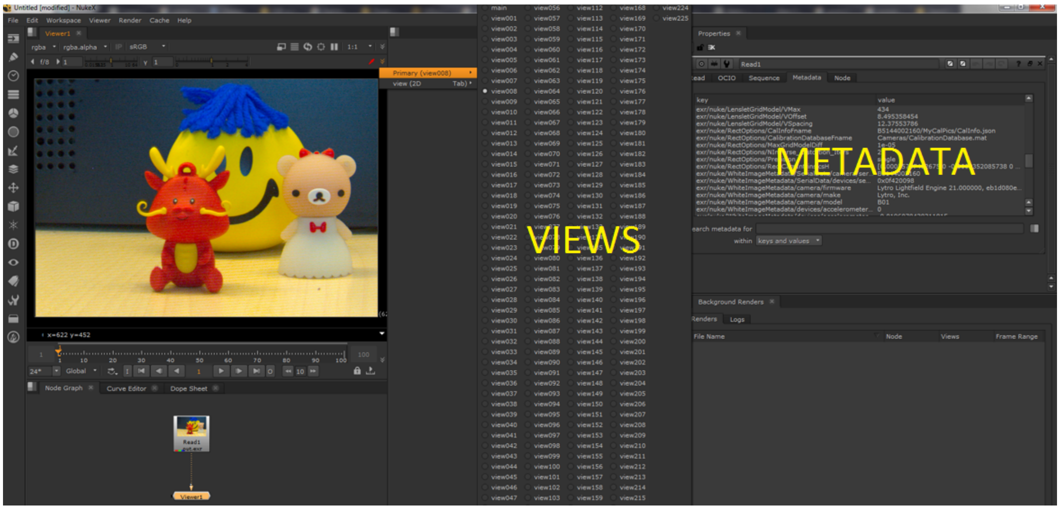
Doc: 794,7K/162,2M

FIGURE 3.9: Light fields transcoded to PSD file format, viewed in Adode Photoshop software

The inverse process reads the layer information from the PSD layer records and forms an output file layout. The image layers are read correspondingly forming the 4D light field structure or the individual camera view structure. The implementation also gathers the relevant metadata from the JSON file and tags it appropriately to the light field image data.

3.5.2 Light Fields <–> OpenEXR

The OpenEXR file format supports storing image layers in two different possibilities, multi-part and multi-view. In multi-part, the sub-aperture images or the camera views can be stored individually, where each view is a standalone EXR file. While, in multi-view, the 2D image assemblies can be embedded together as a single EXR file. As the contrary option is ideal in terms of accessing the data, transmission, storage and also canonical to how the data is dealt as image layers in PSD format, the multi-view option is implemented. The process begins with extracting the non-pixel information and writing the header attributes and its values. The image data are written scan-line based, where the line order begins from top to bottom scan-line. The scan-lines can be randomly accessed and read in different orders. Nevertheless, reading and writing the scan-lines in the same order allows the file to be sequentially read and is time saving.



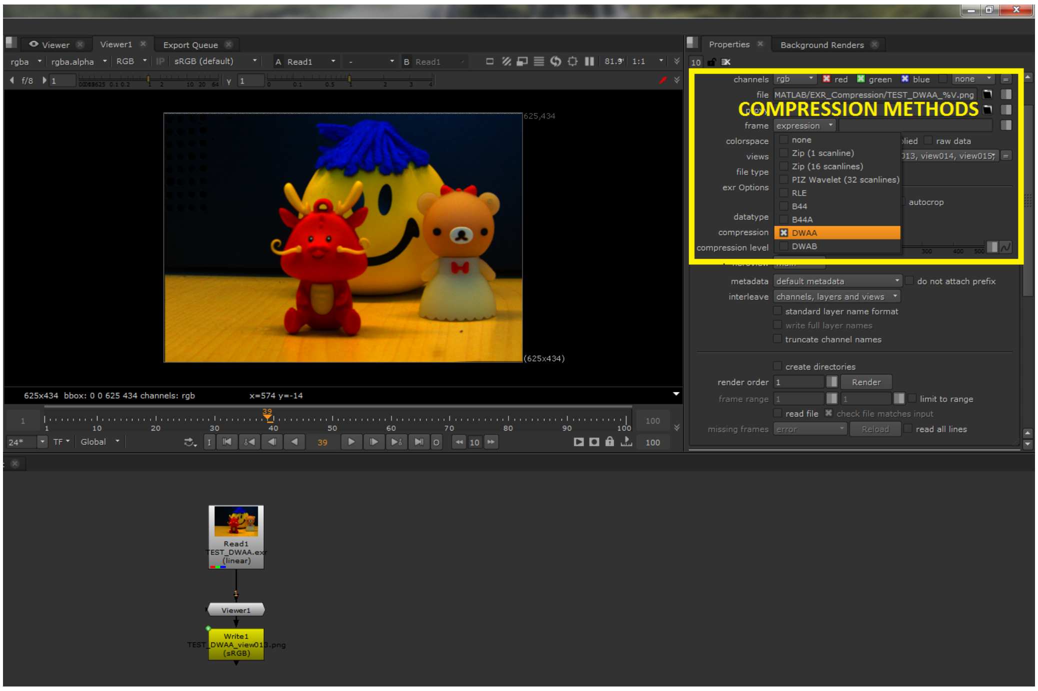
key	value
exr/nuke/LensletGridModel/VMax	434
exr/nuke/LensletGridModel/VOffset	8.495358454
exr/nuke/LensletGridModel/VSpacing	12.37553786
exr/nuke/RectOptions/CalInfoFname	B5144002160/MyCalPics/CalInfo.json
exr/nuke/RectOptions/CalibrationDatabaseFname	Cameras/CalibrationDatabase.mat
exr/nuke/RectOptions/MaxGridModelDiff	1e-05
exr/nuke/RectOptions/N[illegible]	2[illegible]
exr/nuke/RectOptions/Pre[illegible]	[illegible]gle
exr/nuke/RectOptions/Re[illegible]	[illegible]352085738 0 ...
exr/nuke/WhiteImageMeta[illegible]/camera/ser...	B[illegible]160
exr/nuke/WhiteImageMetadata/SerialData/devices/se...	0x0f420098
exr/nuke/WhiteImageMetadata/camera/firmware	Lytro Lightfield Engine 21.000000, eb1d080e...
exr/nuke/WhiteImageMetadata/camera/make	Lytro, Inc.
exr/nuke/WhiteImageMetadata/camera/model	B01
exr/nuke/WhiteImageMetadata/devices/accelerometer...	0

FIGURE 3.10: Light fields transcoded to OpenEXR file format, opened in Nuke software

Unlike in PSD format, the OpenEXR format allows the user to store arbitrary amount of customised metadata fields and of arbitrary type. This provides the flexibility to access the metadata seamlessly within digital compositing platforms like Nuke and GIMP. A light field transcoded to OpenEXR file format and accessed using the Nuke software is illustrated in figure 3.10. The viewer window shows a selected main view in full frame and a list of sub-aperture images as individual views in the side tab. The associated metadata like principle information relating pixels to its corresponding rays and the camera parameters can be effortlessly viewed and accessed from the metadata tab.

3.6 Implication on Post-Processing Algorithms

Both PSD and OpenEXR standards offer various compression options. While PSD file format strictly allows only lossless compression, i.e., Run-Length Encoding (RLE), the OpenEXR file format allows both lossy and lossless compression. For testing and evaluations in this thesis work we will focus in on the OpenEXR file format compression methods. For commonly used texture map images, the lossless compression methods ZIP and ZIPS are preferable. However, for grainy images, PIZ lossless compression is a better choice. For image data containing large areas with similar colors, the RLE lossless compression works efficiently. The lossy compression B44 stores pixel data in HALF color depth, while B44A a variant of the prior mentioned method uses fewer bytes for storing the same pixel data. DWAA and DWAB are powerful lossy compression techniques with adjustable compression levels. The file size can be extremely reduced with higher levels but introduces several artefacts and heavy loss in quality. The preferred and the default value is 45, providing a good trade-off between file size and image quality, with no visible artefacts. The only difference between the two methods is, DWAB compresses blocks of 256 rows at a given time instead of blocks of 32 like in DWAA. All the aforementioned compression schemes can be applied and subjectively analysed using the imaging tools that support OpenEXR files. Figure 3.11 shows the OpenEXR file format supported compression methods directly offered by the Nuke software.



channels rgb ☒ red ☒ green ☒ blue □ none
file MATLAB/EXR_Compression/TEST_DWAA_%V.png
frame expression
compression
□ none
□ Zip (1 scanline)
□ Zip (16 scanlines)
□ PIZ Wavelet (32 scanlines)
□ RLE
□ B44
□ B44A
☒ DWAA
□ DWAB
metadata default metadata □ do not attach prefix
interleave channels, layers and views
□ standard layer name format
□ write full layer names
□ truncate channel names
□ create directories
render order 1
frame range 1 1 □ limit to range
□ read file ☒ check file matches input
missing frames error □ read all lines

FIGURE 3.11: OpenEXR file format supported compression methods offered via the Nuke software

A major reason for light fields to gain traction in VFX and media industry is because of its varied post-processing capabilities. Hence it is essential to evaluate the transcoded and compressed light fields, to investigate if the underlying properties of light fields are intact. For this purpose we have used the light field post-processing applications developed in the V-SENSE [1] project. The denoising [AS17] and super-resolution [AS18] techniques via sparse coding of 4D light fields are tested on the compressed light fields. The EPFL light field dataset [RE16] are deployed for evaluations. In this thesis work the results for the light fields, bush and color chart1, shown in figure 3.12 respectively, are included and discussed.

In addition to the above discussed compression methods, a standard format that supports high bit depth images, JPEG XT is also evaluated on the sub-aperture images of each light field. The state-of-the-art formats JPEG XT [RAE16] and JPEG Pleno [Ebr+16] are designed for high bit depth image data. JPEG XT is an extension to legacy JPEG standard with backward compatibility feature, offering both lossless and lossy representation for HDR images and a legacy text-based encoder for the metadata. JPEG Pleno is an emerging compression standard for newer image modalities such as light

[1] https://v-sense.scss.tcd.ie

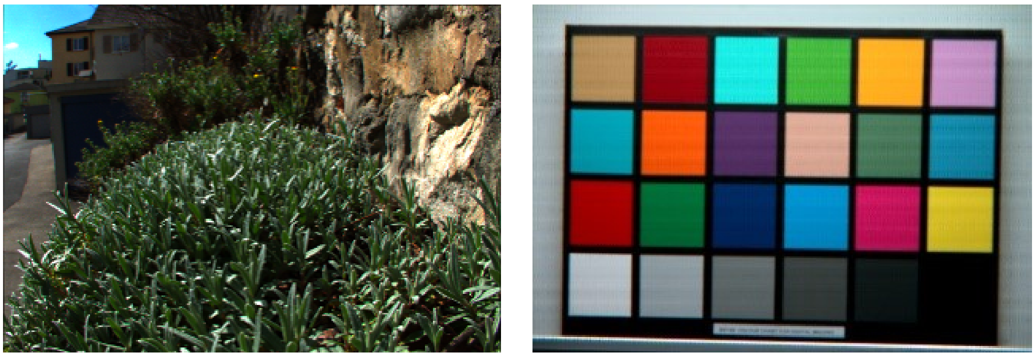


FIGURE 3.12: Light field images used for testing - bush & color chart1

fields, point clouds and holograms, providing comprehensive functionalities to include metadata, allowing image manipulation and interaction [Ebr+16]. More information and our contributions to the JPEG Pleno standard will be discussed in chapter 5.

TABLE 3.1: Comparison of the different compression methods in terms of average PSNR of super-resolution and denoised light field images

Compression Method	**Super-Resolution avg PSNR [dB]**		**Denoising avg PSNR [dB]**	
	Bush	**Color Chart 1**	**Bush**	**Color Chart 1**
None	24.6121	30.4077	34.9620	42.2948
ZIP	24.4785	30.3599	34.9659	42.2913
ZIPS	24.4785	30.3599	34.9696	42.2990
PIZ	24.4785	30.3599	30.4472	42.2966
RLE	24.4785	30.3599	30.4404	42.2858
B44	24.5963	30.2974	30.4027	42.1214
B44A	24.5963	30.2974	30.4024	42.1212
DWAA	24.5038	30.3862	30.4714	42.6126
DWAB	24.6384	30.3862	30.4734	42.6208
JPEG XT	26.7586	31.5601	35.3099	42.3816

Table 3.1 showcases the outcome of evaluating the light field data with super-resolution and denoising techniques. The Peak signal-to-noise ratio (PSNR) is estimated for the individual sub-aperture images of each light field and are averaged respectively and tabulated for the different compression methods. To maintain the synergy between the compression formats, default values are chosen for the lossy methods. Table 3.2 illustrates the compression ratio results. From the recorded values, it can be observed that JPEG XT compression ratio is highly dependent on the image content, while an inverse outcome is obtained in the case of DWAA/DWAB compression. For the color chart1 light field, at a higher compression ratio with DWAA/DWAB

methods, a comparable PSNR value with JPEG XT method is achieved for both denoising and super-resolution. Overall the lossy and lossless methods showcases analogous outcome with both the post-processing techniques and demonstrates that the fundamental properties of the light fields are maintained through the transcoding pipeline and the applied compression methods. Based on the captured scene content and the end user application as a prerequisite, a suitable compression format can be chosen for the light field data.

TABLE 3.2: Comparison of the different compression methods in terms of compression ratio

Compression Method	**Compression Ratio**	
	Bush	**Color Chart 1**
None	1.0000	1.0000
ZIP	1.3468	1.7278
ZIPS	1.3677	1.7533
PIZ	1.3461	1.7644
RLE	1.0150	1.2427
B44	2.2645	2.2860
B44A	2.2785	2.3697
DWAA	2.5913	5.3850
DWAB	2.5972	5.3715
JPEG XT	2.4864	1.6201

TABLE 3.3: Average PSNR and SSIM estimations of the different compression methods

Compression Method	**avg PSNR [dB]**		**avg SSIM**	
	Bush	**Color Chart 1**	**Bush**	**Color Chart 1**
None	89.9006	89.2529	1.0000	1.0000
ZIP	89.9006	89.2529	1.0000	1.0000
ZIPS	89.9006	89.2529	1.0000	1.0000
PIZ	89.9006	89.2529	1.0000	1.0000
RLE	89.9006	89.2529	1.0000	1.0000
B44	49.4720	51.7088	0.9996	0.9997
B44A	49.4720	51.7088	0.9996	0.9997
DWAA	51.5992	52.3635	0.9989	0.9980
DWAB	51.5992	52.3635	0.9989	0.9980
JPEG XT	17.2074	17.9337	0.7686	0.7214

To further validate the compressed light fields the Peak signal-to-noise ratio (PSNR) and the Structural Similarity Index Measure (SSIM) estimations are given in table 3.3. PSNR is computed for each sub-aperture image using the formula:

$$10log_{10}(\frac{R^2}{MSE})$$

where $R = 65,535$, the maximum possible pixel value of an image and the estimations are averaged over all the views. SSIM quantifies the degradation caused in the image due to image processing steps such as data compression or loss of data during transmission. The computed SSIM values are averaged over the respective light fields. When observed, the evaluations are interesting compared to the PSNR values from table 3.1 of super-resolution and denoising. As described in [RAE16], the image encoding in JPEG XT standard is based on a RGBE two-layer image format. The data reduction is performed by first transforming the image into a tone-mapped version and then a reconstructive multiplier image is stored. Analyzing and viewing these images with conventional software can exclude the multiplier image, enabling the viewer to see only the tone-mapped version represented in a standard dynamic range. The post-processing techniques indeed produce competitive results of the compressed light fields.

Figure 3.13 showcases the difference images for the bush light field image. A reference view is compared with differently compressed images resulting after super-resolution. The resulting images are amplified and cropped uniformly as shown in figure 3.13 a), b) and c) for no compression, DWAB compression and JPEG XT compression respectively. The two post-processing techniques used for evaluations from the V-SENSE project are patch based. The algorithm applies a search window over the angular and spatial views to reconstruct a 5D patch for each reference patch. As most compression methods employ a low-pass filter and because of the high frequencies, the denoising and super-resolution algorithms tend to use incorrect patches, introducing artefacts, reducing the PSNR. Once compression is performed, these patches have lower frequencies and fit appropriately, thereby increasing the overall PSNR estimations. We also verified and confirmed this from the patch table generated by the post-processing algorithms on the sub-aperture images prior to and after compression. The no compression (i.e., using raw sensor data) and lossless compression methods showcased analogous patch selections in comparison to the lossy compression methods and the JPEG XT format.

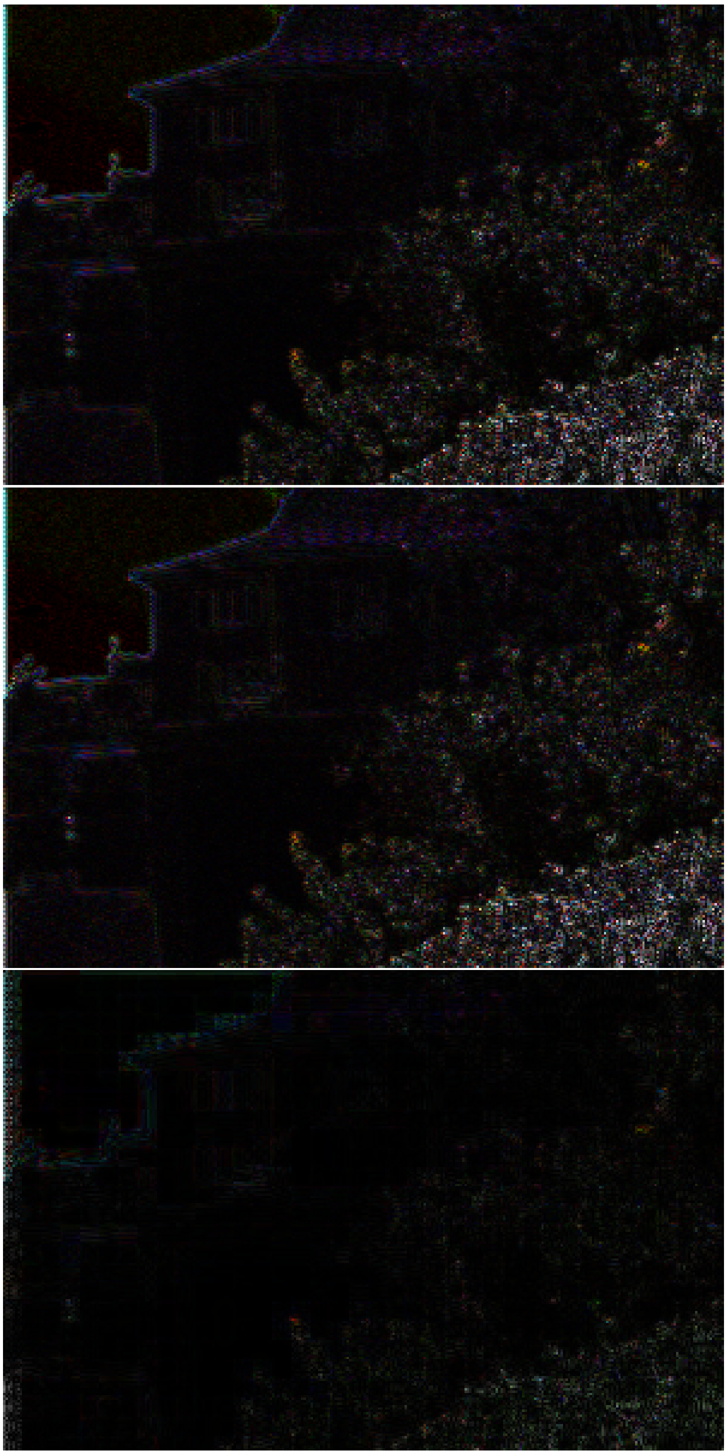


FIGURE 3.13: Image differencing between the given reference frame and the respective super resolutioned frame for Bush light field image (amplified and cropped uniformly)
a) no compression; b) DWAB; c) JPEG XT

Chapter 4

Optimized Sequencing of Light Field Data

Light fields are representations of light and its direction in three-dimensional space. By capturing light fields, a large amount of data is recorded for characterizing a scene in 3D compared to conventional imagery recording only color intensities. The increasing amount of post-processing capabilities acquired through light field imagery is accompanied by the demand for additional memory and storage requirements. This in turn implies the need for higher data rates on services and devices used in exchange, broadcast and display of light fields. Hence, compression of light field imagery is a fundamental step to facilitate the use of light field data with the current professional and consumer applications. In this chapter, the approaches we have implemented and evaluated to optimise light field data to be compatible with the state-of-the-art image and video codecs are discussed in detail.

4.1 Pseudo-Temporal Reordering

This section deals with a low-complexity light field compression technique by pre-processing the sub-aperture images pseudo-temporally and adapting them to the state-of-the-art codecs. Our proceedings related to this work can be referred to in [P8] and [P4]. In accordance with the literature, some of the interesting research works include data compression of light fields by disparity compensated prediction [MG00] and compression for rendering of light fields [Cha+04]. Improved results have been proposed by reordering the light field data prior to coding using H.264/AVC [FK05], using HEVC [Vie+15], by slicing the lenslet array 2D image [PA16], by tiling the sequence [PG17], with multi-view coding structure [Liu+16] and as well lossless compression of light fields [Per15]. Nevertheless, most of the techniques fail to fully exploit the correlation between the sub-aperture images efficiently. By pseudo-sequencing the light field frames using our proposed idea, the results have consistently outperformed in comparison with still image codecs and also other state-of-the-art reordering approaches.

4.1.1 The Proposed Layout

The steps of the proposed encoding layout are shown in 4.1. The raw sensor data are fed into the image decoding pipeline and we obtain the rectified and color-corrected assemblies of 2D images, which represents a stack of consecutive conventional images from different perspectives. As light fields recorded using handheld plenoptic cameras have a restricted viewing angle, the perspective change between the sub-aperture images is narrow and consequently is well correlated. Hence, the redundancy among the different views can be utilized for inter frame prediction with an optimal reordering approach.

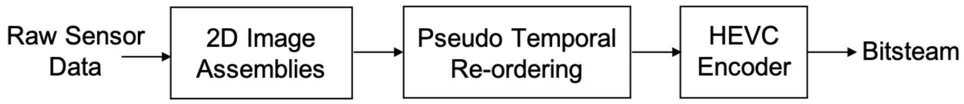


FIGURE 4.1: Encoding layout

From observing the individual sub-aperture images in figure 4.2 a) of the decoded light field introduced in chapter 2, it can be seen due to the interpolation of the data from a hexagonal raster (the shape of the lenslets in the lenslet array) to a rectangular layout, some of the corner views comprise of less or almost no useful information of the scene. The objective of the re-reordering technique in this context is to ensure grouping of the less competent views consecutively, to avoid generating higher residuals while predicting the adjacent views.

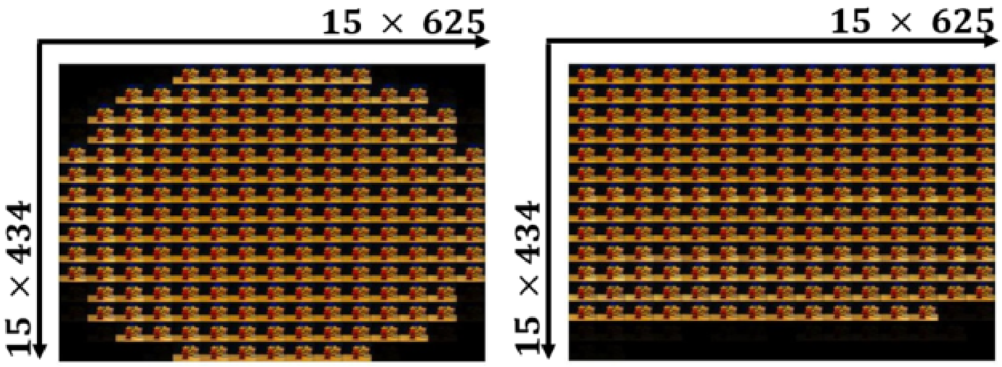


FIGURE 4.2: Concatenated frames
a) as generated by the Matlab light field toolbox;
b) circularly reordered

In general the well-known reordering techniques like linewise figure 4.3 a), tiling figure 4.3 b), and zigzag figure 4.3 c) are widely used in data compression. As, these approaches fail to match the requirement of resorting the unusable corner views together, we have introduced the pixelated circular reordering technique, figure 4.3 d). With our technique we achieve both, arranging the neighbouring views that exhibit high spatial redundancy together and the corner views are sorted consecutively towards the end, as illustrated in figure 4.2 b).

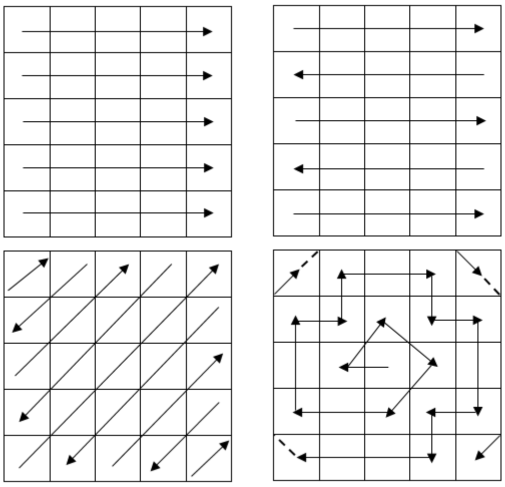
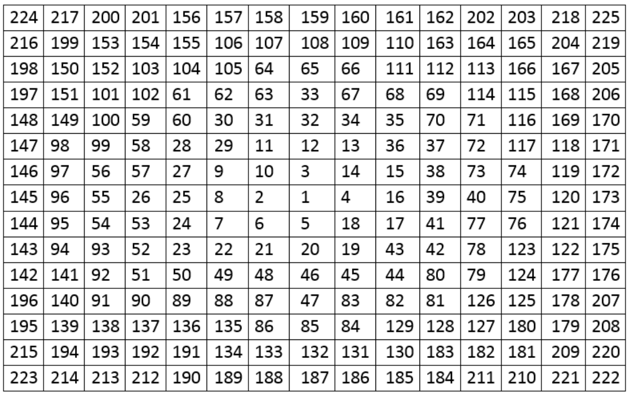


FIGURE 4.3: Reordering Approaches; From top left to bottom right - a) linewise; b) tiling; c) zigzag; d) circular (proposed)

The pattern generated for pseudo-temporally reordering the light field sub-aperture images recorded using the Illum camera can be seen in figure 4.4. The matrix represents the frame numbering in which the views are sorted and composed into a video stream to be compatible for compression with the state-of-the-art video codecs. Our algorithm is scalable to generate pseudo-temporal sequences for any desired number of sub-aperture images, i.e angular views.



224	217	200	201	156	157	158	159	160	161	162	202	203	218	225
216	199	153	154	155	106	107	108	109	110	163	164	165	204	219
198	150	152	103	104	105	64	65	66	111	112	113	166	167	205
197	151	101	102	61	62	63	33	67	68	69	114	115	168	206
148	149	100	59	60	30	31	32	34	35	70	71	116	169	170
147	98	99	58	28	29	11	12	13	36	37	72	117	118	171
146	97	56	57	27	9	10	3	14	15	38	73	74	119	172
145	96	55	26	25	8	2	1	4	16	39	40	75	120	173
144	95	54	53	24	7	6	5	18	17	41	77	76	121	174
143	94	93	52	23	22	21	20	19	43	42	78	123	122	175
142	141	92	51	50	49	48	46	45	44	80	79	124	177	176
196	140	91	90	89	88	87	47	83	82	81	126	125	178	207
195	139	138	137	136	135	86	85	84	129	128	127	180	179	208
215	194	193	192	191	134	133	132	131	130	183	182	181	209	220
223	214	213	212	190	189	188	187	186	185	184	211	210	221	222

FIGURE 4.4: Proposed pseudo-temporal reordering matrix for 15 x 15 view layout (225 frames)

4.1.2 Analysis with Image and Video Codecs

The sub-aperture images are evaluated with both image and video compression standards. The experiments are conducted for the bit rates 0.1bpp, 0.25bpp, 0.5bpp and 1bpp respectively. All the images are sampled at 4:2:0.

For the still image compression standards, JPEG [Int88] and JPEG 2000 [ITU02], the quality factor parameter is varied to yield the desired bit-rate. For the HEVC [Sul+12] video coding standard, the Quantization Parameter (QP) is adapted to achieve the desired bit-rate and the low-delay predictive main configuration is used for encoding. We have also evaluated the HEVC intra coding, by varying the quantization parameter.

Images from the EPFL light field dataset [RE16] are used for experiments. Six different light field images, shown in figure 4.5 with diverse content are chosen to showcase the efficiency of our approach. The compressed images are evaluated in terms of objective quality using both Peak signal-to-noise ratio (PSNR) and Structural Similarity Index Measure (SSIM).

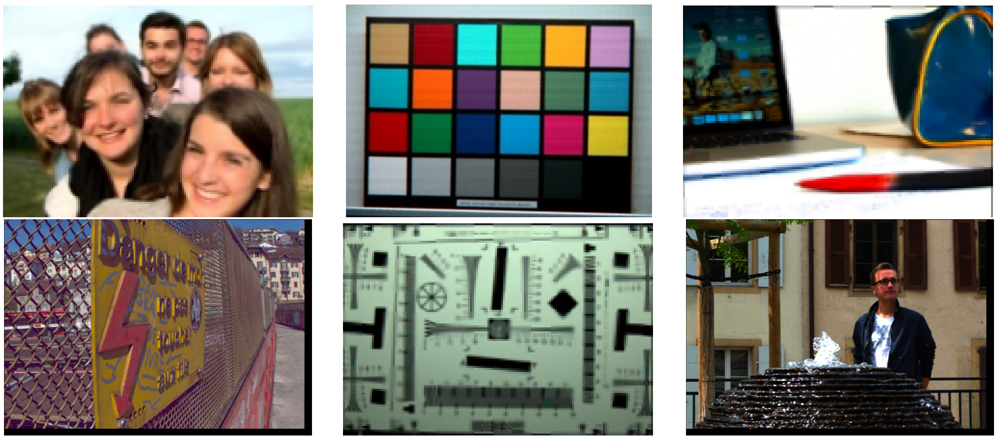


FIGURE 4.5: Light field images used for analysis; From top left to bottom right - a) Friends; b) Color Chart; c) Desktop; d) Danger de Mort; e) ISO Chart; f) Fountain & Vincent2

The results of encoding the light field images with JPEG in comparison to the proposed pseudo-temporal sequencing with HEVC [Sul+12] is exhibited in figure 4.6. At first glance, it can be observed that for all tested compression ratios, the rate-distortion performance of the recommended approach outperforms the JPEG compression. Primarily, as a result of pseudo-sequencing the frames using the proposed technique re-sorts the light field content in an efficient way, which supports in high coding performance. The pseudo-temporal sequencing facilitates in exploiting the redundancies in both spatial and angular domain by the HEVC codec, compared to only exploiting the spatial redundancies by JPEG. An increase of up to 22 dB gain over JPEG encoding is achieved.

Figure 4.7 illustrates the rate-distortion results between the well-known reordering techniques such as linewise, tiling and zigzag against the proposed pseudo-circular reordering. The improvements are steady across different light field images and for all tested compression ratios. An increase of up to 1 dB gain is achieved with the proposed technique compared to other reordering approaches. This again substantiates that the proposed pseudo-temporal

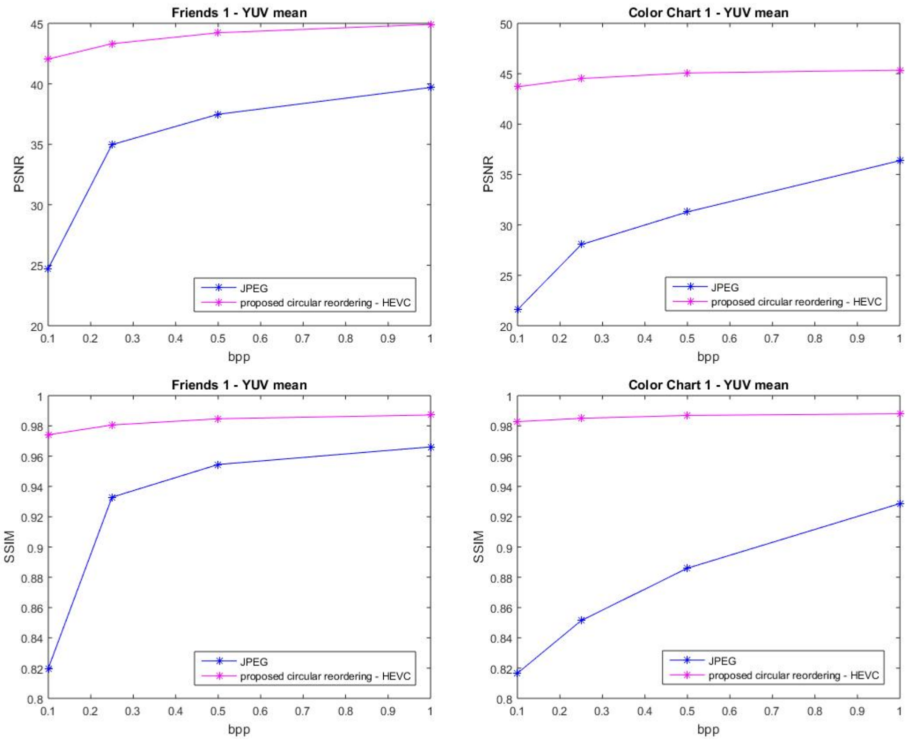


FIGURE 4.6: PSNR & SSIM vs bpp - comparing JPEG still image compression to proposed pseudo-temporal reordering using HEVC compression

sequencing approach maximizes the correlation between the neighbouring frames and the state-of-the-art video codecs are able to better exploit the data redundancy by both intra and inter-frame prediction.

TABLE 4.1: PSNR values comparing JPEG, JPEG 2000, HEVC Intra and HEVC for different bit rates

	Friends				Color Chart				Danger de Mort			
PSNR [dB]	**0.1 bpp**	**0.25 bpp**	**0.5 bpp**	**1 bpp**	**0.1 bpp**	**0.25 bpp**	**0.5 bpp**	**1 bpp**	**0.1 bpp**	**0.25 bpp**	**0.5 bpp**	**1 bpp**
JPEG	24.7147	34.9692	37.4815	39.6955	22.0048	28.8525	32.1829	36.5629	23.6042	30.8431	34.0714	37.3709
JPEG 2000	34.9570	37.9544	40.9406	44.2627	38.0949	41.6276	44.0908	46.8767	29.6621	32.5135	35.2525	38.8852
HEVC Intra	34.3110	37.5710	40.6801	43.3137	42.1788	43.7241	44.5103	45.1946	28.7790	31.5073	33.4346	36.8825
HEVC	42.0462	43.3121	44.2190	44.9146	43.8919	44.7161	45.2655	45.5961	36.0970	37.9440	38.9998	40.4753

TABLE 4.2: SSIM values comparing JPEG, JPEG 2000, HEVC Intra and HEVC for different bit rates

	Friends				Color Chart				Danger de Mort			
SSIM	**0.1 bpp**	**0.25 bpp**	**0.5 bpp**	**1 bpp**	**0.1 bpp**	**0.25 bpp**	**0.5 bpp**	**1 bpp**	**0.1 bpp**	**0.25 bpp**	**0.5 bpp**	**1 bpp**
JPEG	0.8199	0.9330	0.9546	0.9662	0.8252	0.8591	0.8905	0.9295	0.6834	0.8572	0.9220	0.9499
JPEG 2000	0.9002	0.9322	0.9584	0.9791	0.9530	0.9705	0.9794	0.9871	0.7992	0.8691	0.9349	0.9640
HEVC Intra	0.9130	0.9445	0.9660	0.9804	0.9791	0.9827	0.9845	0.9869	0.7825	0.8563	0.8947	0.9428
HEVC	0.9741	0.9806	0.9847	0.9872	0.9830	0.9852	0.9872	0.9883	0.9342	0.9538	0.9629	0.9732

Table 4.1 and 4.2 showcases results comparing the different image and video codec's rate-distortion performance in terms of PSNR and SSIM at different

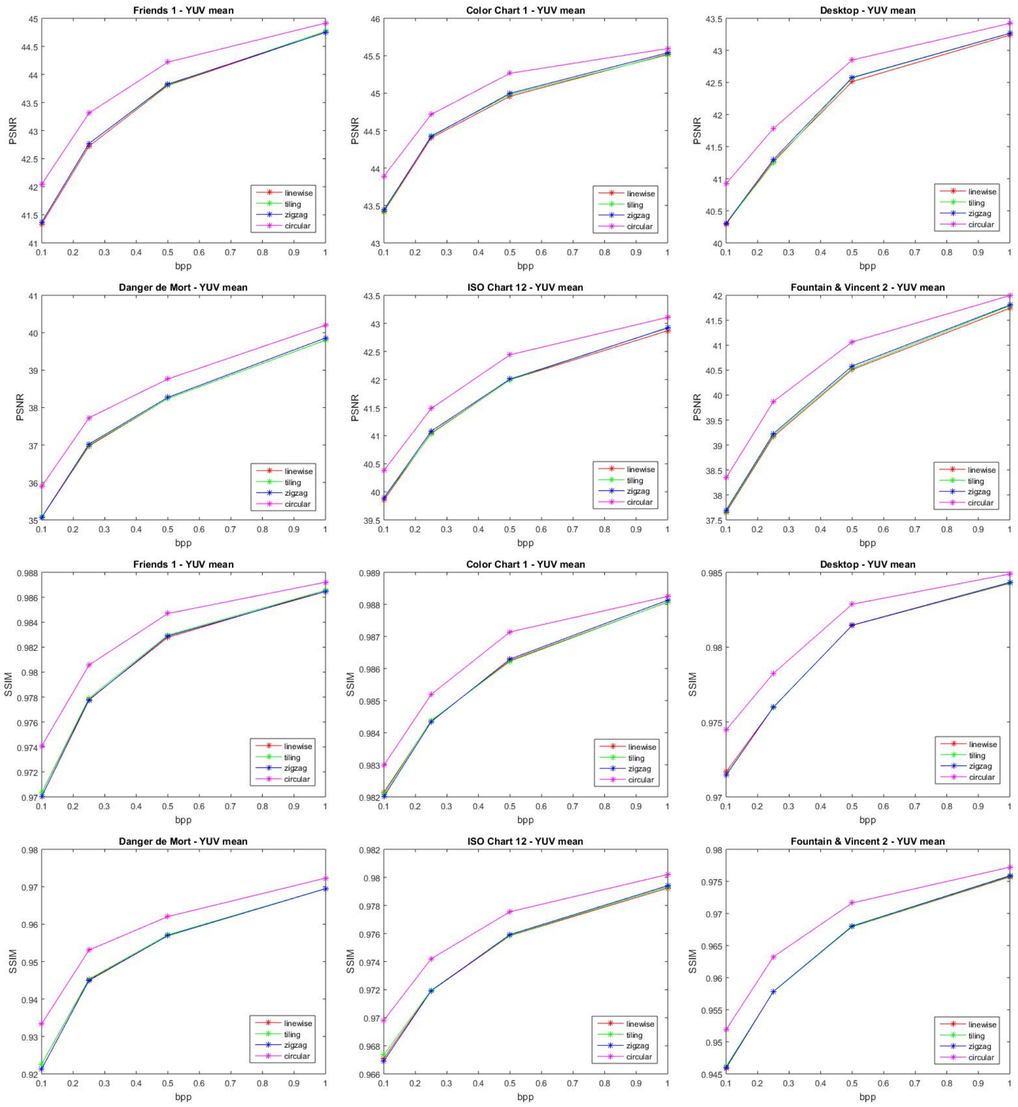


FIGURE 4.7: PSNR & SSIM vs bpp - analysis of the different reordering techniques (linewise, tiling and zigzag) in comparison to the proposed pseudo-temporal reordering using HEVC compression

bit rates. The outcome demonstrates JPEG considerably under-performing compared to other still image codecs, JPEG 2000 and HEVC intra prediction. On the contrary, observing the curves in figure 4.8, JPEG 2000 and HEVC intra competes closely and difference is performance is in the range of 0.0-1.0 dB for the different bit rates. Also, another interesting observation from the rate-distortion curves is, HEVC exhibits a larger difference in performance for higher compression ratio, i.e., 0.1bpp but the difference is considerably reduced for lower compression ratio, i.e., 1bpp. It can be realised that the advantages of exploiting the spatial and angular correlation reduces for lower compression ratios. For such lower compression ratios, utilizing only the spatial redundancy would yield comparable coding efficiency than making use of both spatial and angular dimensions. However, with our proposed pseudo-temporal reordering approach it is possible to fully exploit the inherent properties of the video encoders, obtaining higher compression ratio. Another implicit advantage is, the recommended method do not introduce changes to the codec. Hence, it can be applied to all standard video codecs and seamlessly integrated to the available storage and broadcast services.

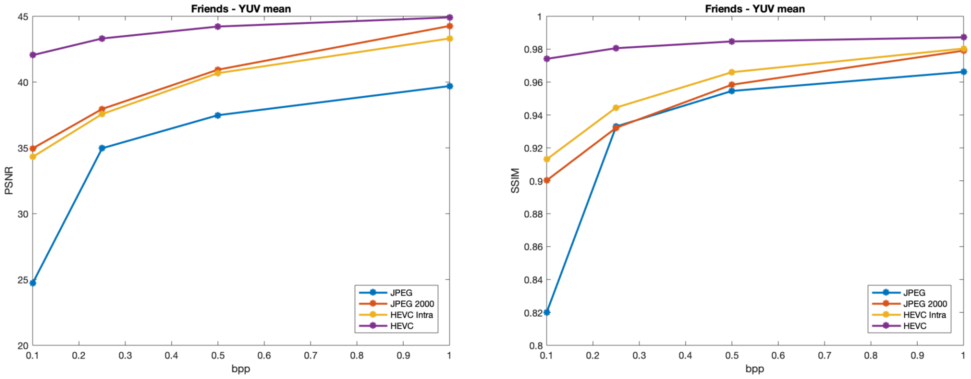


FIGURE 4.8: Rate Distortion curves comparing JPEG , JPEG 2000, HEVC Intra and HEVC for Friends; a) PSNR vs bpp and b) SSIM vs bpp

4.2 Pre-processing with Adaptive Gaussian Filtering based Superpixels

This section evolves on the integration of light fields into the state-of-the-art image and video processing chains by pre-processing the light field data with superpixel segmentation based adaptive Gaussian filtering and pseudo-temporal sequencing. Pre-processing of video data is a widely-known approach to achieve bit rate reduction in compressed bit streams. Most of the techniques exploit the temporal and spatial correlation. By filtering an image within the Just Noticeable Distortion (JND) threshold [CL95], the human eye fails to perceive the changes owing to the limitations in the human visual system. A state-of-the-art research work on video pre-processing showcases the reduction of bit rate by utilizing the perceptual redundancy in the video data [Din+15]. This is achieved using superpixel based image segmentation, along with JND based Gaussian filtering. A fundamental step in image processing is to segment objects of interest in an image automatically. In superpixel segmentation, pixels with similar attributes such as color, texture or brightness are grouped into clusters segmenting the image into meaningful parts. Gaussian filtering of images is a commonly employed technique in image compression for reducing noise in the images and to produce smooth transitions between pixel intensities.

The proposed layout is as shown in figure 4.9. The essential initial step is to decode the raw sensor data and produce the 2D image assemblies. In the

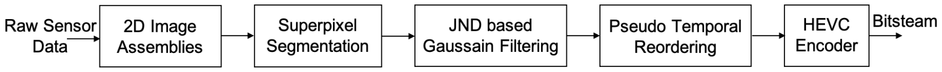


FIGURE 4.9: Proposed pre-processing layout

following steps, the sub-aperture images are independently segmented into superpixels and the superpixels are Gaussian filtered with the JND threshold. Finally the filtered frames are pseudo-temporally reordered before encoding. The core implementation steps are explained in detail below.

4.2.1 Superpixel Segmentation

By perceptually grouping pixels in an image, superpixel clusters are formed, resulting in over-segmentation of an image. Compared to rectangular patches, superpixels align well with the edges and object boundaries in an image and carry more information than single pixels. The superpixel segmentation algorithms can be categorised either as gradient ascent or graph based methods. The gradient ascent algorithms are recursive, starting with a random clustering of pixels then iteratively combining the clusters until a given threshold is reached. Some of the techniques are, watershed in digital spaces [VVS91], mean shift algorithm [CM02], quick shift kernel method [VS08], turbopixels [Lev+09] and Simple Linear Iterative Clustering (SLIC) [Ach+10]. On the other hand graph based methods exploit the cost function between the neighbouring pixels. The pixels are considered as nodes and the edges connecting the pixel nodes have weights, which is used for clustering. Few of the graph based algorithms are graphcut textures [Kwa+03], [FH04], superpixel lattices [Moo+08] and supervoxels [VBM10]. Each of the above mentioned algorithms perform both qualitatively and quantitatively different. In general, a superpixel segmentation algorithm must function robust, consuming less memory and the resulting superpixels must adhere well to the object contours. We in our previous work have implemented an enhanced SLIC algorithm [P1] and [P2] that outperforms in terms of both boundary recall and compactness. The state-of-the-art SLIC algorithm is a k-means clustering approach, segmenting an image based on color information. The enhanced SLIC algorithm can employ additional information beyond color channels to obtain more accurate and meaningful segmentation. A dynamic calculation of weights is introduced, which eliminates the complexity of deciding the weighting parameters of the different channels. User input is reduced to only the image to be segmented and the desired number of superpixel segments.

In the second step of the proposed layout, the sub-aperture images are segmented into superpixels k. The size of each superpixel S is of dimension sxs and in this research work we chose the size as 16x16 pixels, same as the size of a macroblock. Considering the consecutive step, where the individual superpixels are adaptively Gaussian filtered, the recommended superpixel size is propitious for intra-frame prediction. For a sub-aperture image of resolution

(x, y), the number of superpixel clusters $k = round(\frac{(x,y)}{256})$. With the feature of dynamic distance calculation, the distance D is measured as in equation 4.1. The weighting parameters m, n are calculated for the spatial proximity D_S and the color proximity D_C respectively.

$$D = \sqrt{m.D_S^2 + n.D_C^2} \tag{4.1}$$

The weighting factors are determined with the function given in equation 4.2, which uses the number of edge pixels e in every superpixel S.

$$f(e,S) = \frac{69(2.25ln(\frac{2200e}{S^2}+1) - 2.25ln(\frac{2200}{S^2}+1))}{((2.25ln((\frac{2200*10^6}{S^2}+1) - 2.25ln(\frac{2200}{S^2}+1))} + 1 \tag{4.2}$$

The function $f(e,S)$ formulated to determine weighting values is a derivative of the μ-law [Skl88]. As the derivations and the background algorithm details are beyond the scope of this dissertation, check our works [P1] and [P2] for an overview. The resulting superpixel segmentation of the light field image Desktop (centre SAI) using the proposed technique is shown in figure 4.10.

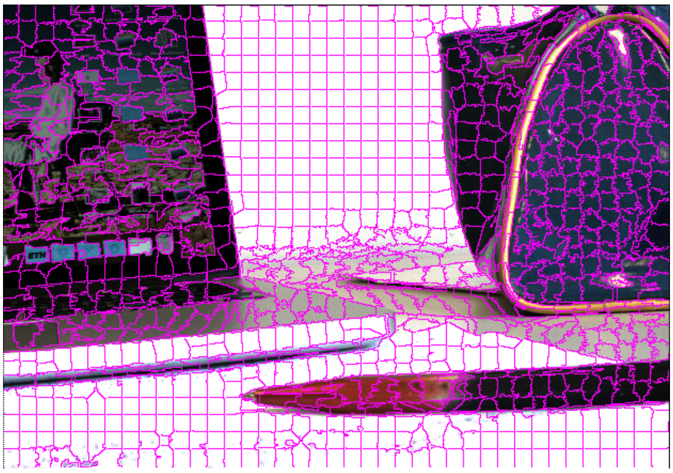


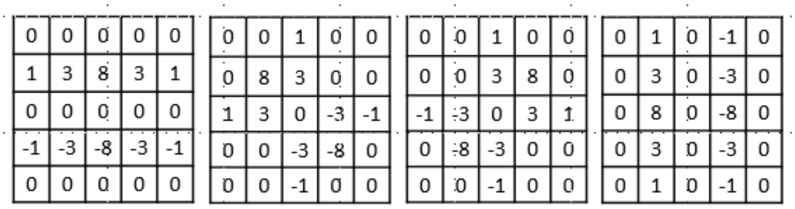
FIGURE 4.10: Superpixel segmentation of Desktop light field image using enhanced SLIC

4.2.2 JND based Gaussian Filtering

The third phase of the layout is filtering the superpixels with JND based Gaussian parameters. The JND threshold is determined from the texture and gradient around each pixel (xy) for each sub-aperture image.

$$G_f(x,y) = \frac{1}{16}\sum_{i=1}^{5}\sum_{j=1}^{5} Img(x-3+i, y-3+j) * g_f(i,j) \tag{4.3}$$

The weighted normal of luminance changes $G_f(x,y)$ around every pixel is determined as demonstrated in equation 4.3, utilizing the four filters g_1, g_2, g_3, g_4 as illustrated in figure 4.11. Then, as in equation 4.4, the maximum weighted average of gradients around every pixel (xy) is determined, with a maximum of four $g(f = 1,2,3,4)$ filters. Figure 4.12 a) displays the corresponding reconstructed image of the center view from the desktop scene.



0	0	0	0	0
1	3	8	3	1
0	0	0	0	0
-1	-3	-8	-3	-1
0	0	0	0	0

0	0	1	0	0
0	8	3	0	0
1	3	0	-3	-1
0	0	-3	-8	0
0	0	-1	0	0

0	0	1	0	0
0	0	3	8	0
-1	-3	0	3	1
0	-8	-3	0	0
0	0	-1	0	0

0	1	0	-1	0
0	3	0	-3	0
0	8	0	-8	0
0	3	0	-3	0
0	1	0	-1	0

FIGURE 4.11: The filters g1, g2, g3, g4 used for calculating the weighted average of luminance changes

$$G_{max}(x,y) = \max_{f=1,2,3,4}\{|G_f(x,y)|\} \tag{4.4}$$

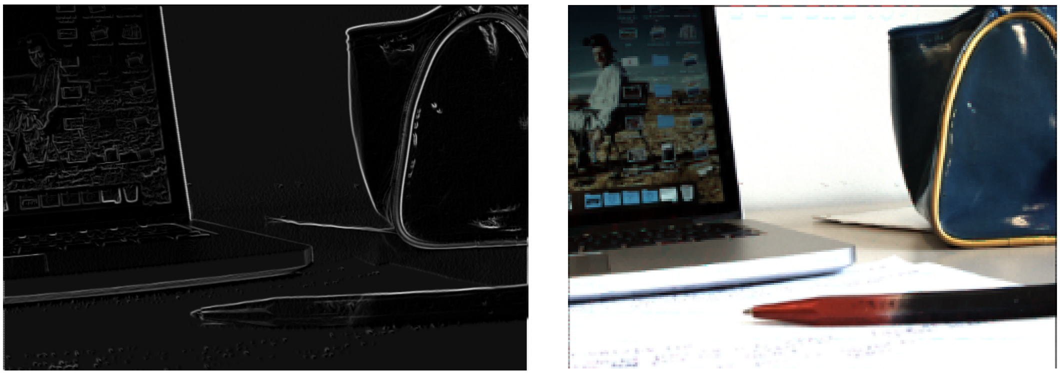


FIGURE 4.12: Center view of the Desktop light field image; a) Reconstructed view after maximum weighted average of luminance; b) JND based Gaussian filtered view

$$\sigma = \begin{cases} 8.5 - 0.5 * avgG_{max}(S), & avgG_{max}(S) < 15 \\ 1, & 15 \leq avgG_{max}(S) \leq 30 \\ 0, & avgG_{max}(S) > 30 \end{cases} \tag{4.5}$$

Filtering an entire frame with the same Gaussian parameter often results is a major loss of visual quality. Accordingly, each of the superpixel is filtered with adaptive Gaussian parameters. Initially, the mean of the $avgG_{max}(S)$ maximum weighted gradient average of all pixels within each S superpixel is determined. Then, as shown in equation 4.5, the σ standard deviation value for a Gaussian filter is determined. Figure 4.12 b) displays the JND dependent Gaussian filtered image of the center view from the desktop scene.

4.2.3 Comparison and Results

The final pre-processing stage is reordering the JND based Gaussian filtered sub-aperture images pseudo-temporally. As proposed in the previous section 4.1, by circularly reordering the frames, the neighbouring frames exhibit high correlation and the frames with least information are grouped sequentially together. As in dense light fields, the perspective variations between the adjacent views are very minimal the reordering facilitates the inter frame prediction process where the redundancy between the frames are fully exploited.

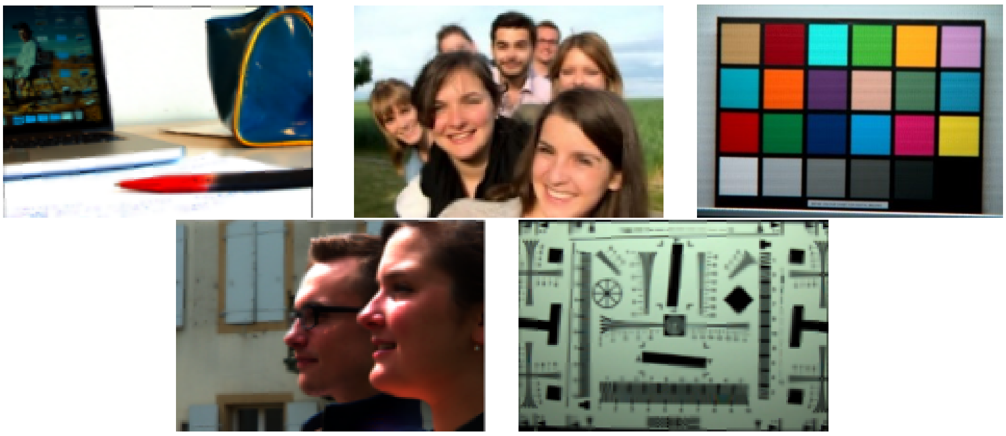


FIGURE 4.13: Light field images used for analysis; From top left to bottom right - a) Desktop; b) Friends; c) Color Chart; d) Sophie & Vincent; e) ISO Chart

The proposed pre-processing technique is accessed in terms of both perceived visual quality and bite-rate reduction. Images from the EPFL dataset [RE16] are used for testing. We have chosen five different images, as shown in figure 4.13 with versatile content to showcase the robustness in terms of determining the filtering parameters and its outcome.

For evaluations, different quantization values over a range are considered, QP - 20, 24, 28 and 32 respectively. The reduction in bit-rate of the proposed pre-processing layout against only pseudo-temporally reordering the light field data are compared. To maintain the synergy, in both cases HEVC codec with the configuration of low delay predictive main is employed for coding. The outcomes are classified in table 4.3 for the different light field images in accordance with the quantization parameters. It is evident that for all the light field images, the recommended pre-handling technique has surpassed and a maximum average bit-rate reduction of 21.86% is achieved for the color chart light field image. As the color chart contains several color blocks of uniform intensity and less textures, the parameter selection and filtering has worked efficiently eliminating the high frequencies.

TABLE 4.3: Bit rate reduction using the proposed technique in comparison to pseudo-temporally reordered data bit stream

Image / **Rate**	**I-01**	**I-02**	**I-03**	**I-04**	**I-05**
QP 1	-19.94%	-23.94%	-13.80%	-6.74%	-19.26%
QP 2	-19.72%	-16.37%	-27.74%	-4.98%	-14.99%
QP 3	-13.58%	-7.68%	-27.15%	-2.42%	-7.93%
QP 4	-8.63%	-2.19%	-18.74%	-0.49%	-3.20%
Average	**-15.47%**	**-12.55%**	**-21.86%**	**-3.66%**	**-11.35%**

The perceived visual quality of the compressed light fields are analysed in figure 4.14 for the desktop image. The center sub-aperture image is showcased in three variants, uncompressed and compressed at QP values 20 and 32 respectively. Upon intently comparing the images (especially in a high resolution monitor), the loss in visual quality is almost negligible, while the resulting light fields compressed with higher quantization values are relatively smoother. Critically, as the most intriguing attributes of light fields are the post-processing prospects, software refocusing of the compressed light fields are as well investigated. As exhibited in figure 4.15, the center views are consistently refocused at a given depth (precisely, at pixel position [323,417]). From observing the images, it is evident that the refocusing results are as expected artefacts or loss in quality. Thereby, our approach is ideal for adapting the light field data for the available state-of-the-art video codecs without imposing any changes to the reference implementation.

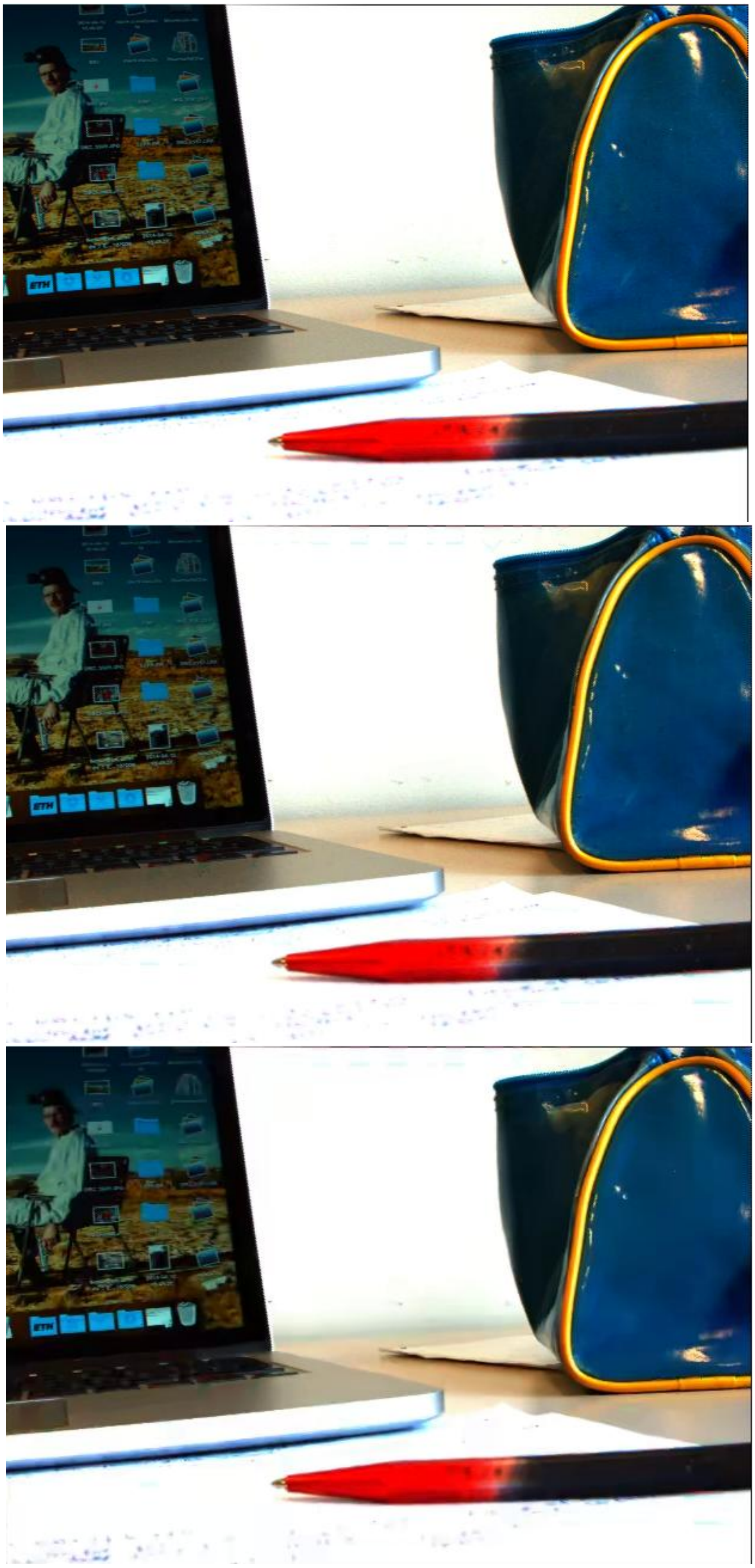


FIGURE 4.14: All-in focus views - a) uncompressed; b) compressed with QP-20; c) compressed with QP-32

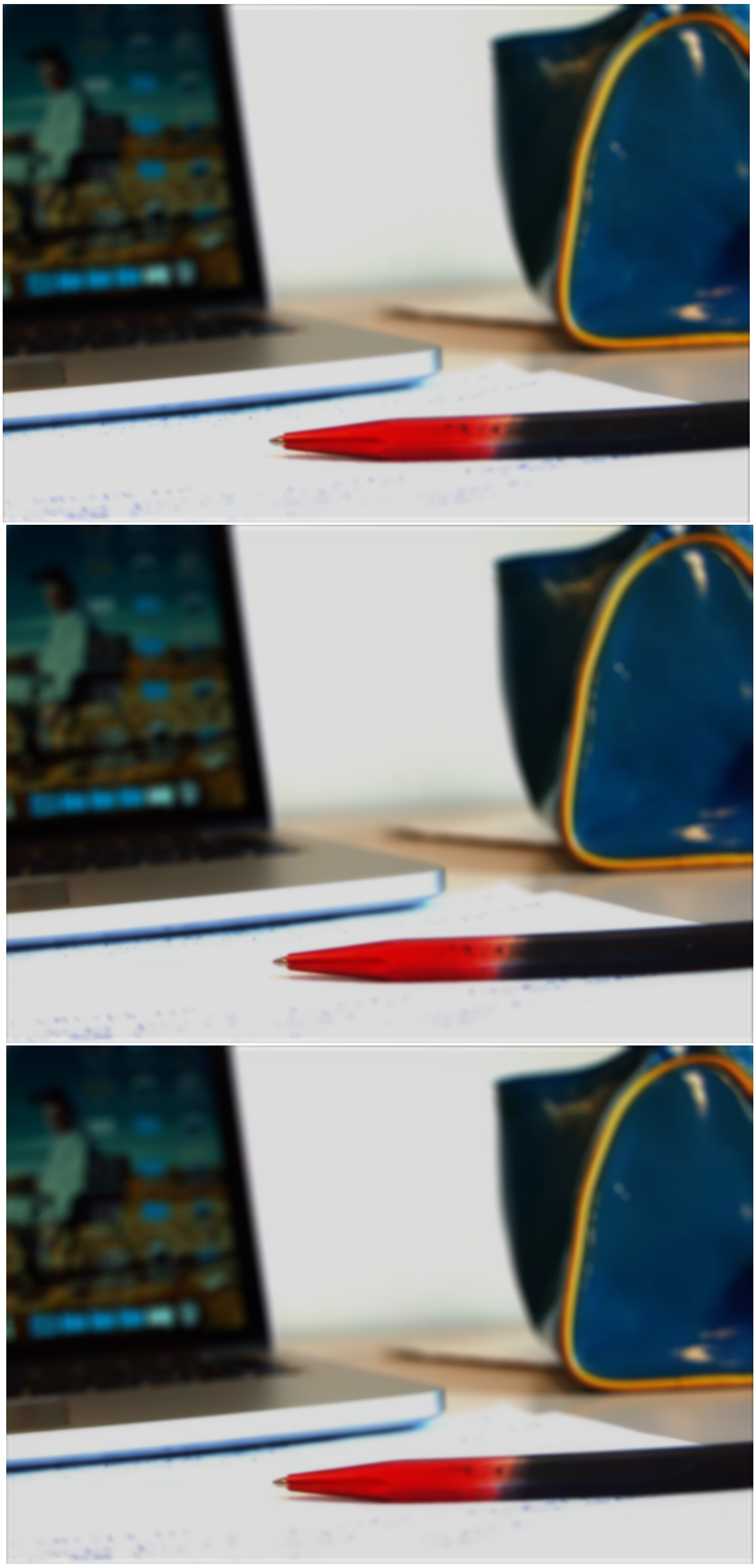


FIGURE 4.15: Refocused views - a) uncompressed; b) compressed with QP-20; c) compressed with QP-32

4.3 Proximity Maximizer

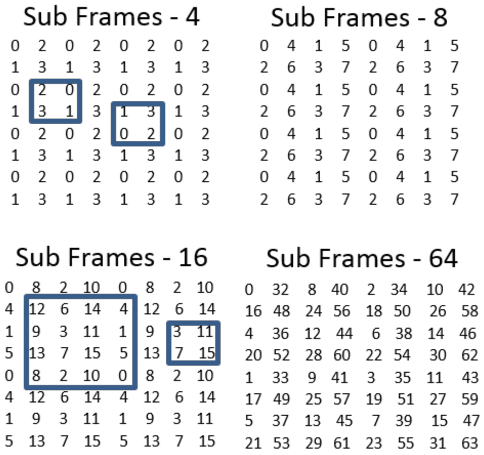
The number of dimensions representing light has once again increased, including the time domain. 4D light fields captured with additional temporal information per ray or as assemblies of rays include the 5^{th} dimension, namely time and thus produce 5D light fields. The related research works, including our works mentioned in the above sections have paved way to ideas on efficient 4D light field compression. However, techniques for compression and storage for higher dimensions is still an open challenge. In this section we have introduced a predictive coding approach of 5D light fields by automatic generation of per frame customized coding structure exploiting both spatial and temporal neighbors. This is very crucial when we have moving objects in the scene.

Apart from related works discussed in the previous sections, an important group of research techniques reorders the light field sub-aperture images in pseudo-temporal sequence and encodes them using the standard codecs achieving high prediction efficiency [Con+18]. The works, LF-TSP [Ima+19] and optimized reference picture selection [Mon+19] are similar to our proposed idea, where the technique operate under the same methodology, reordering the frames and adapting the reference lists. While these state-of-the-art methods are optimized and tested on dense lenslet based 4D light fields and have achieved good results, our proposed method includes the additional temporal domain with novel 5D light field sub-framing patterns. Our technique offers an efficient pre-processing technique to overcome the challenges imposed by additional dimensions and adapt the light field data for the available state-of-the-art codecs.

4.3.1 5D Light Fields

The 5D light field camera array [P9], introduced in section 2.2.2.2 consists of 8x8 synchronized cameras arranged with constant distances both vertical and horizontal. Images are generated at 40fps with a resolution of 1920x1200. The rig is electronically controlled and the different cameras can be configured to trigger at varied time instances enabling the temporal behavior. We have recorded the HaToy dataset [HLC19] using the 5D light field camera array.

The HaToy scene, shown in figure 3.4 incorporates several static and moving components of variable sizes and complex geometry. All the objects in the scene are made visible in all the cameras and they fully capture the static and moving parts of the scene. The dataset includes several spatio-temporal capturing patterns as illustrated in figure 4.16 in addition to the uniform synchronized capturing. These unique sub-framing patterns are derived using two-dimensional bit reversal permutation procedure described in figure 4.17. In figure 4.16, from the highlighted regions, it can be seen that neighboring cameras have different phases and the phases are equidistantly distributed



Sub Frames - 4

0	2	0	2	0	2	0	2
1	3	1	3	1	3	1	3
0	2	0	2	0	2	0	2
1	3	1	3	1	3	1	3
0	2	0	2	0	2	0	2
1	3	1	3	1	3	1	3
0	2	0	2	0	2	0	2
1	3	1	3	1	3	1	3

Sub Frames - 8

0	4	1	5	0	4	1	5
2	6	3	7	2	6	3	7
0	4	1	5	0	4	1	5
2	6	3	7	2	6	3	7
0	4	1	5	0	4	1	5
2	6	3	7	2	6	3	7
0	4	1	5	0	4	1	5
2	6	3	7	2	6	3	7

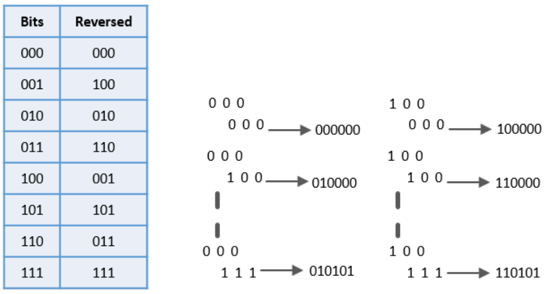
Sub Frames - 16

0	8	2	10	0	8	2	10
4	12	6	14	4	12	6	14
1	9	3	11	1	9	3	11
5	13	7	15	5	13	7	15
0	8	2	10	0	8	2	10
4	12	6	14	4	12	6	14
1	9	3	11	1	9	3	11
5	13	7	15	5	13	7	15

Sub Frames - 64

0	32	8	40	2	34	10	42
16	48	24	56	18	50	26	58
4	36	12	44	6	38	14	46
20	52	28	60	22	54	30	62
1	33	9	41	3	35	11	43
17	49	25	57	19	51	27	59
5	37	13	45	7	39	15	47
21	53	29	61	23	55	31	63

FIGURE 4.16: Bit reversal sub-framing



Bits	Reversed
000	000
001	100
010	010
011	110
100	001
101	101
110	011
111	111

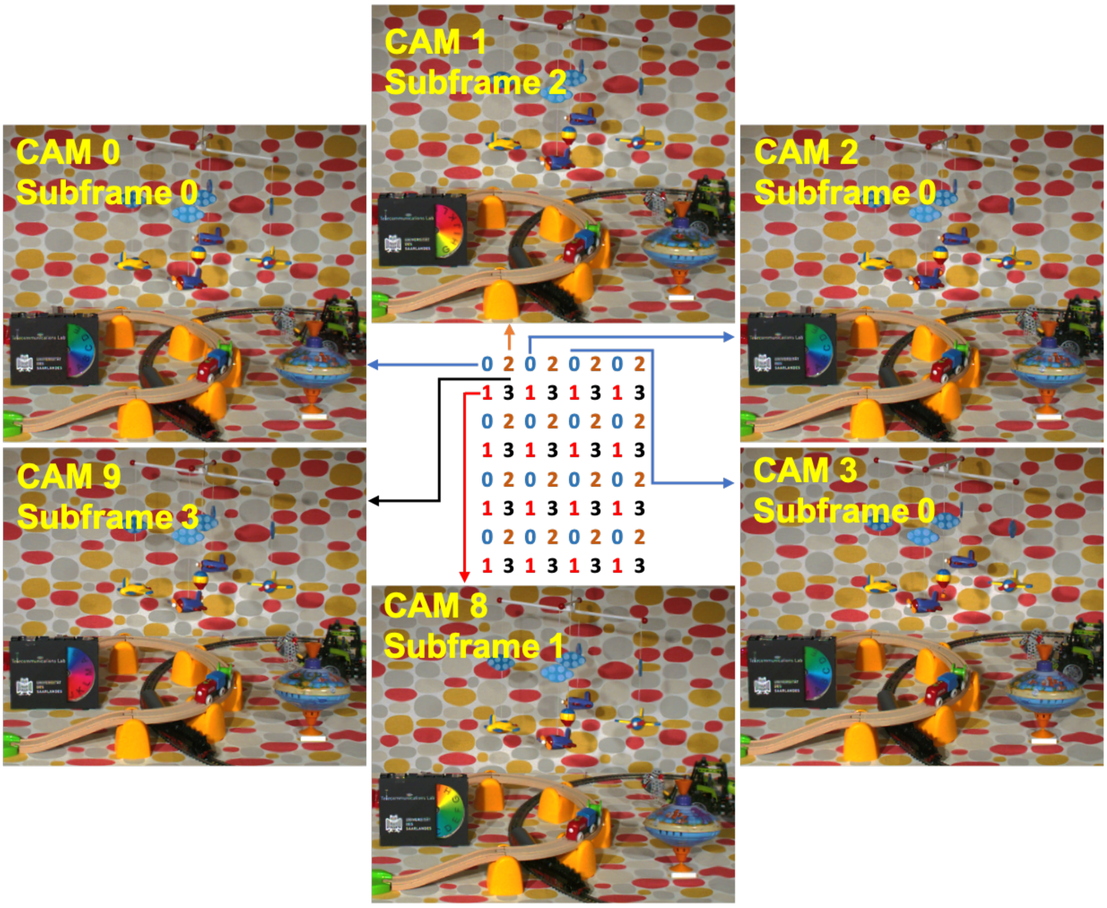
FIGURE 4.17: Bit reversal procedure

within the layout.

With respect to the 5DLF representation, we have t=0:1:N, whereby, 0..N-1 belongs to one full frame and hence the spacing is 1/(N*40) seconds (or 25ms/N). a and b are the camera indices from 0..7, but for intuitive understanding we have the camera numbering, from top left (0) to bottom right (63). Before predicting the HaToy sub-aperture images, the most interesting objects to consider are the fast spinning ones like the CD drive and the spin top. From figure 4.18, we can observe that only parts of these objects' texture are visible on each camera. For sub-framing by a factor of 4 the center cameras #27,28,35,36 respectively stem from four different sub-frames #3,1,2,0 and hence have a high temporal resolution for moving and a high angular resolution for static parts of the scene, while cameras #18,20,34,36 all stem from the same sub-frame #0 and hence are angular neighbors for moving parts of the scene. The temporal behavior is significant and has to be considered while predicting the sub-frames.

4.3.2 Processing Pipeline

The proposed mechanism for integration of light field data into standard video coding chains is as shown in figure 4.19. The core of the processing



0 2 0 2 0 2 0 2
1 3 1 3 1 3 1 3
0 2 0 2 0 2 0 2
1 3 1 3 1 3 1 3
0 2 0 2 0 2 0 2
1 3 1 3 1 3 1 3
0 2 0 2 0 2 0 2
1 3 1 3 1 3 1 3

FIGURE 4.18: Understanding the 5D HaToy dataset

pipeline is the proximity maximizer implementation, which generates the reordering layout based on the camera array setup and the user desired start position. The pseudo-code of the algorithm is as follows.

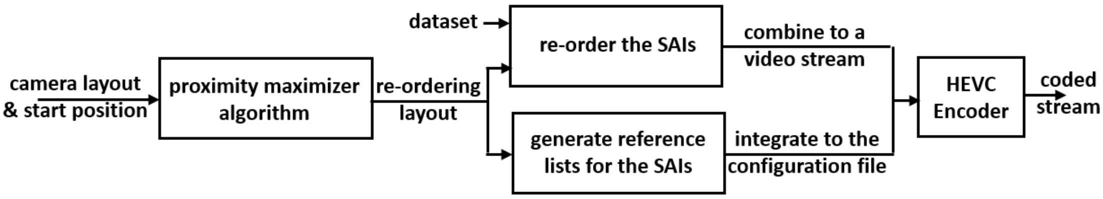


FIGURE 4.19: Processing Pipeline

Algorithm 1: Proximity Maximizer

```
Algorithm proximity_maximizer($RowSize, ColSize, StartPosition$)
    Output: $ReorderLayoutMatrix$
    Data: // Global variables initialised only once
    $BUFFER\_SIZE \leftarrow 15$
    Initialization:
    $buffer \leftarrow FIFOQueue(BUFFER\_SIZE)$
    $UnvisitedCells \leftarrow \{1 : RowSize * ColSize\}$
    $ResultOrder = \{\}$
    $CurrentPosition = StartPosition$
    while $UnvisitedCells \neq \{\}$ do
        $UnvisitedCells \leftarrow UnvisitedCells - \{CurrentPosition\}$
        $ResultOrder \leftarrow ResultOrder \cup \{CurrentPosition\}$
        $buffer.Push(CurrentPosition)$
        $CurrentPosition \leftarrow GetNextPosition()$

Procedure GetNextPosition()
    $NextPosition := -1$
    $MaxNeighborCount := -1$
    foreach $Cell \in UnvisitedCells$ do
        $NeightborCount \leftarrow GetNeighborCount(Cell)$
        if $NeightborCount > MaxNeighborCount$ then
            $NextPosition \leftarrow Cell$
            $MaxNeighborCount \leftarrow NeightborCount$
    return $NextPosition$

Procedure GetNeighborCount($Cell$)
    $r, c := GetRowCol(Cell)$ // Get the row and col of the cell
    $Rows \leftarrow \{r\} \cup \{r-1 \iff r > 1\} \cup \{r+1 \iff r < RowSize\}$
    $Cols \leftarrow \{c\} \cup \{c-1 \iff c > 1\} \cup \{c+1 \iff c < ColSize\}$
    $NeighbourCount := 0$
    foreach $r\_ind \in Rows$ do
        foreach $c\_ind \in Cols$ do
            $CellVal := CellValueAt(r\_ind, c\_ind)$
            if $CellVal \in buffer$ then
                $NeighbourCount := NeighbourCount + 1$
    return $NeighbourCount$
```

Predicting a sequence of images works ideal when the image to be predicted has immediate neighbors that are already coded and are available for prediction in the picture buffer. Hence, a sequence which maximizes the availability of processed cells (aka. image from a camera position) during prediction of a new cell is desirable. This forms the motivation of the proximity maximizer algorithm. The algorithm tries to maximize the neighbor count by greedily selecting the next optimal cell location to be predicted in the sequence having the maximum neighbors. The process starts with a cell position selected by the user as the key SAI, hence the user has the possibility to decide which parts of the scene would be covered by the key frames. Then the algorithm finds the next cell in traversal path by searching through all unvisited cells, for the one with the maximum number of neighbors. If multiple cells are found with the same number of maximum neighbors, the first position is considered. This functionality is achieved by the GetNextPosition procedure which in turn calls the GetNeighbourCount to compute the number of cells available with in the immediate proximity. Then the best found candidate is positioned in the reordering layout. The process continues until all cells are optimally repositioned. The state-of-the-art video codecs imposes an upper bound over the size of the Decoded Picture Buffer (MaxDpbSize) to a maximum of 16 frames (in our case sub-aperture images), which includes 15 previously coded SAIs, available as references + the current SAI, to be predicted. However, most of the decoders and media players operate appropriately on a much lower number of active reference frames, with maximum of 8 as the limitation. Taking these scenarios into consideration we have also included the degree of freedom in the algorithm to decide the queue size for potential neighbours, to not to overflow the buffer limit of the reference pictures for prediction.

The reordering layout is then utilized for reordering the SAIs. Once the SAIs are reordered, they are combined into a video stream which forms the input for the codec. The picture reference lists for the coding structure are generated with the last eight frames for prediction. This also aids in overcoming the downside in few cases where the algorithm looks only at its immediate neighbors and suffers from the local maxima problem while finding the best path. Then the complete coding structure is generated for the desired Group of Pictures (GOP). The coding structure is integrated into the overall configuration file which is used for coding the input video stream. The streams are predictively coded with the HEVC reference implementation.

4.3.3 Evaluation

The reordering layout exhibited in table 4.4 is generated for a 8x8 camera setup configuration, used in capturing the HaToy dataset and one of the center positions is chosen as the desired starting cell. The resulting pseudo-temporal sequence exhibits high correlation between consecutive frames and maximizes the proximity of immediately available neighbors for prediction, thereby exploiting utmost redundancy. One of the advantage of this proposed algorithmic reordering technique, compared to the previous works with fixed prior reordering, is the flexibility of building the layout depending on the camera grid and also selecting the desired key frame to construct the pseudo reordering sequence around it. This ensures that the algorithm can be adapted to the different camera grid layouts, depending on the capturing setup and the user has the possibility to select the key frames with the most scene content for prediction.

TABLE 4.4: Sample reordering layout for an 8x8 camera setup

13	**10**	**9**	**11**	**45**	**44**	**46**	**48**
12	**8**	**5**	**6**	**42**	**41**	**43**	**47**
14	**7**	**2**	**3**	**39**	**38**	**40**	**49**
15	**16**	**4**	**1**	**34**	**35**	**37**	**50**
17	**18**	**19**	**27**	**30**	**33**	**36**	**51**
21	**20**	**22**	**26**	**29**	**32**	**52**	**53**
24	**23**	**25**	**28**	**31**	**56**	**54**	**55**
64	**63**	**62**	**61**	**60**	**58**	**57**	**59**

Table 4.5, summarizes the experimental results in which the scanning order used in the previous research methods and our proposed method are compared for the different sub-framing. Several intermediate quantization parameter values from the available range [0-51] are tested to analyze finer to coarser levels of quantization. Mean YUV-PSNR is calculated for the proposed technique against row-wise default scanning and pseudo-temporally re-ordered scanning. It can be observed that for all compression ratios, the proposed pre-processing technique has outperformed the other sortings and for higher QP values, a gain of more than 2dB PSNR is achieved. As mentioned in the literature work on Data Compression [Sal07], with just an increase of 0.5dB PSNR, the magnitude of improvement is already visible to the human eye. From the results it can be observed for coarser quantization the differences are more or less quantized towards zero and hence large quantization parameters directly reflect the quality of prediction. By algorithmically resorting the candidates based on the scene we directly influence the prediction and achieve maximum prediction gain.

Additional experimental results comparing our technique with the most currently deployed state-of-the-art image codec HEIC are exhibited in table 4.6. Evaluations are carried out with HEVC in intra-mode, where the SAIs are coded independently of each other. For fairness in the comparison, quality analysis are performed on similar datarates for both the codecs. As we observe the Rate Distortion (R-D) performance illustrated in figure 4.20, the increase in gain is prominent for the proposed method over HEIC for all compression ratios. The fact that by pseudo-video reordering using the proposed mechanism organizes the 5D light field data in an optimal sequence exploiting the redundancies between SAIs, resulting in higher coding efficiency. For further validation, examining the compressed SAIs show evident distortions in the scene parts with the fast moving objects like the CD drive and the spin top. Figure 4.21 shows an original SAI and the same SAI predicted at 0.1bpp using HEIC and HEVC. It can be seen that the prints on the spin top are highly distorted when the SAI is intra predicted using HEIC compared to HEVC which uses the neighboring SAIs as references, maximizing the prediction gain.

TABLE 4.5: YUV - PSNR for the different capturing patterns

YUV –PSNR [dB]	Uniform Frames			
	Default	Circular Scanning	Proposed Method	Gain c4 – c2
QP 40	35.3060	36.6862	37.7064	2.4004
QP 36	37.5960	38.8210	39.9239	2.3279
QP 32	40.1016	41.0969	42.1196	2.0180
QP 28	42.4384	43.0640	43.9549	1.5165
QP 24	44.6091	44.9291	45.6846	1.0755
QP 20	46.5197	46.7158	47.3315	0.8118
QP 16	48.5068	48.5947	49.0751	0.5683

YUV –PSNR [dB]	4 Subframes			
	Default	Circular Scanning	Proposed	Gain c4 – c2
QP 40	35.3907	36.5640	37.7034	2.3127
QP 36	37.6798	38.8994	39.9076	2.2278
QP 32	40.1841	41.0912	42.1035	1.9194
QP 28	42.5257	43.0597	43.9409	1.4152
QP 24	44.6540	44.9547	45.6977	1.0437
QP 20	46.5249	46.7423	47.3325	0.8076
QP 16	48.5407	48.5989	49.0793	0.5386

YUV –PSNR [dB]	8 Subframes			
	Default	Circular Sacnning	Proposed	Gain c4 – c2
QP 40	35.4092	36.5638	37.7121	2.3029
QP 36	37.6826	38.8109	39.9199	2.2373
QP 32	40.2046	41.0906	42.1089	1.9043
QP 28	42.5274	43.0623	43.9490	1.4216
QP 24	44.6892	44.9528	45.7038	1.0146
QP 20	46.5331	46.7225	47.3352	0.8021
QP 16	48.5479	48.6010	49.0801	0.5322

TABLE 4.6: YUV – PSNR for HEIC vs HEVC proposed

YUV - PSNR [dB]	Uniform Frames				4 Subframes				4 Subframes			
	0.1 bpp	0.25 bpp	0.5 bpp	1 bpp	0.1bpp	0.25 bpp	0.5 bpp	1 bpp	0.1 bpp	0.25 bpp	0.5 bpp	1 bpp
HEIC	41.1043	46.2101	48.8097	51.2331	41.1067	46.2114	48.8119	51.2358	41.1056	46.2153	48.8127	51.2352
HEVC	46.1337	48.2014	49.5731	51.7140	46.1252	48.2017	49.5708	51.7147	46.1303	48.2043	49.5665	51.7117
Gain	5.0294	1.9913	0.7634	0.4809	5.0185	1.9903	0.7589	0.4789	5.0247	1.9890	0.7538	0.4765

Hence our proposed approach facilitates the integration of light fields into standard video processing chains by algorithmically re-ordering the image data to maximize the prediction gain. A PSNR gain of more than 2dB is achieved with the HEVC codec, purely from the prediction technique.

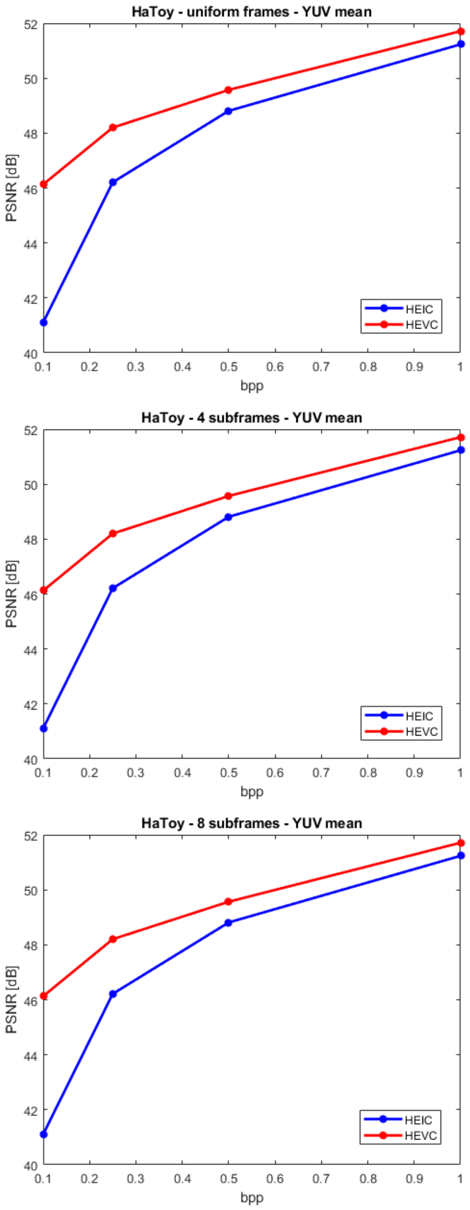


FIGURE 4.20: Rate distortion results of HEIC and the proposed technique with HEVC

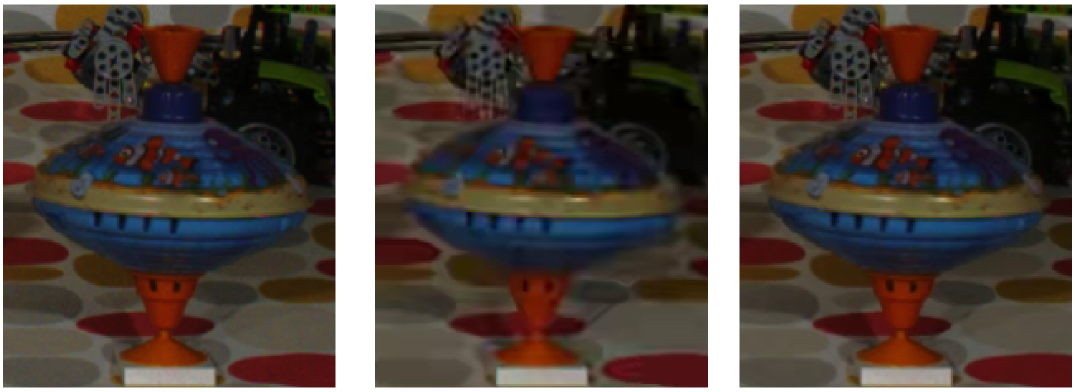


FIGURE 4.21: A sample SAI (zoomed and cropped uniformly) a) original; b) HEIC predicted; c) proposed HEVC predicted

Chapter 5

Light Field Coding Framework

With the advent of different plenoptic modalities, the challenges towards interoperability between devices and applications has increased, furthermore at a cross-modality level. In consideration to these challenges the JPEG and the MPEG standardization committees have initiated new standards, JPEG Pleno and MPEG-I respectively. JPEG Pleno [Ast+20] supports the plenoptic modalities such as light fields, point clouds and holographic representations. MPEG-I [Dom+17] is a collection of specifications to digitally represent immersive media, both multi-dimensional audio and video representations. With respect to the timeline of this thesis write-up, both the standards are still under-development.

The contributions in this chapter are focused and limited towards supporting the standardization committees with a novel light field dataset. The available light field datasets are limited to 4D and 4.5D assets, offering only still images and synchronous videos. As members of the JPEG committee and based on the interests from the Pleno sub-group, we designed the HaToy scene. The HaToy scene consists objects of various sizes and complex geometry, incorporating multiple static and independently moving components. The scene is captured using our 5D light field camera array. The RAW data is adapted to the test requirements and is shared with the community to extend and evaluate light field coding solutions to not only synchronous light field assets but also to sub-framed light field assets.

5.1 JPEG Pleno

JPEG Pleno is an emerging standard that aims to support newer light representations such as point clouds [Sch+18], light fields [Ebr+16] and holographic imaging [Sch+19]. The standard is being exclusively designed to code all modalities of plenoptic functions and provide full functionality to include metadata, image manipulation and interaction. As an appointed member of the JPEG group (the national representative – DIN [1]), we had the opportunity to participate and contribute in several group meetings. Although the international standard is yet to be published, as members of the committee, we could access and actively experiment with the Pleno VM (Verification Model). Based on the interests from the JPEG Pleno group, we designed and

[1] https://www.din.de/de

captured the HaToy dataset, introduced in figure 4.18, captured using the 5D light field camera array 2.2.2.2.

For coding light fields, Pleno offers two independent and conceptually different codecs [Per+19]. The MULE codec performs transform coding and works efficiently only on lenslet light fields [de +20]. The WaSP codec uses prediction coding and is an option for both lenslet and camera array images (dense and sparse light fields) [AT19b]. It is a depth based/disparity compensated prediction, which uses occlusion-aware depth estimation. The codec utilizes normalized disparity for warping the reference SAIs to the target SAIs and the prediction works efficiently for planar camera configurations. For very dense datasets the warping is performed hierarchically based on least-squares method.

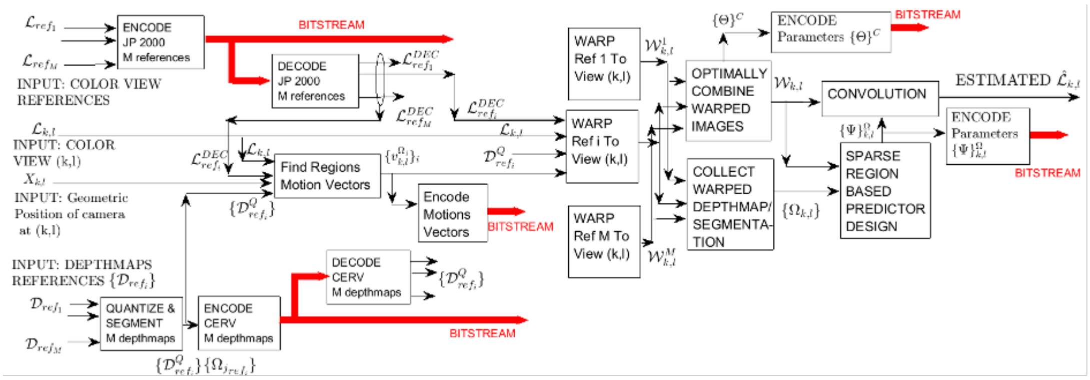

$\mathcal{L}_{ref_1}$
$\mathcal{L}_{ref_M}$
ENCODE JP 2000 M references
BITSTREAM
INPUT: COLOR VIEW REFERENCES
DECODE JP 2000 M references
$\mathcal{L}^{DEC}_{ref_1}$
$\mathcal{L}^{DEC}_{ref_M}$
$\mathcal{L}^{DEC}_{ref_i}$
$\mathcal{L}_{k,l}$
INPUT: COLOR VIEW (k,l)
$X_{k,l}$
INPUT: Geometric Position of camera at (k,l)
Find Regions Motion Vectors
$\{v^{\Omega_i}_{k,l}\}_i$
$\mathcal{D}^Q_{ref_i}$
$\{\mathcal{D}^Q_{ref_i}\}$
Encode Motions Vectors
BITSTREAM
INPUT: DEPTHMAPS REFERENCES $\{\mathcal{D}_{ref_i}\}$
$\mathcal{D}_{ref_1}$
$\mathcal{D}_{ref_M}$
QUANTIZE & SEGMENT M depthmaps
ENCODE CERV M depthmaps
DECODE CERV M depthmaps
BITSTREAM
$\{\mathcal{D}^Q_{ref_i}\}\{\Omega_{j_{ref_i}}\}$
WARP Ref 1 To View (k,l)
WARP Ref i To View (k,l)
WARP Ref M To View (k,l)
$\mathcal{W}^1_{k,l}$
$\mathcal{W}^M_{k,l}$
$\{\Theta\}^C$
ENCODE Parameters $\{\Theta\}^C$
BITSTREAM
OPTIMALLY COMBINE WARPED IMAGES
$\mathcal{W}_{k,l}$
CONVOLUTION
ESTIMATED $\hat{\mathcal{L}}_{k,l}$
$\{\Psi\}^{\Omega}_{k,l}$
ENCODE Parameters $\{\Psi\}^{\Omega}_{k,l}$
BITSTREAM
COLLECT WARPED DEPTHMAP/ SEGMENTATION
$\{\Omega_{k,l}\}$
SPARSE REGION BASED PREDICTOR DESIGN

FIGURE 5.1: Block diagram of the WaSP encoder
Figure illustrated from [AT18]

Astola *et al* in their works [AT19a], [AT19b] have described the functional blocks of the codec as following. The primary operation of the WaSP codec is to predict each target view based on the reference views, as illustrated in the block diagram 5.1. The codec requires high quality depth estimation as it is necessary to efficiently reconstruct the side views. For each reference, its corresponding disparities, both horizontal and vertical, anchored at the side view is obtained by warping its reciprocal depth. Then a disparity refinement process using motion vectors is performed to overcome scaling issues regarding vertical and horizontal baselines. To merge the warped reference views, the least-squares view merging is performed, thereby minimizing the sum of residuals for every pixel. The last prediction stage is used to find which of the regressors are required in a prediction template to perform final convolution of the merged warped image with a sparse predictor.

As WaSP encoder requires both texture and disparity information of the SAIs for prediction, we adapted the HaToy texture maps in PPM format and generated the disparity maps compatible to the Pleno specification. Due to HaToy's complex scene geometry, the state-of-the-art depth/disparity map algorithms [Hon+17], [RSM20] and [CAS20] failed to produce quality disparity

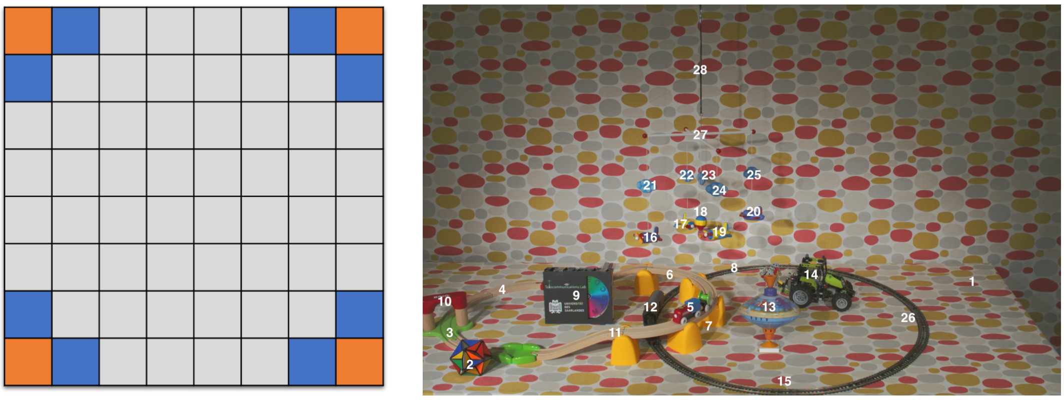


FIGURE 5.2: a) Frame pairs (orange camera positions - target frames; blue camera positions - reference frames); b) Objects tags

maps. Following which, a human-assisted semi-automatic technique for disparity map generation was implemented. Adjacent camera pair images, as illustrated in figure 5.2 a) (orange camera positions - target frames; blue camera positions - reference frames) were annotated manually to obtain characteristic feature points, as recorded and showcased in figure 5.3 (for the target frame 007, using the adjacent frames 006, 015 respectively) at varied depths over several objects, indicated using the different object tags, shown in 5.2 b). These characteristic points are used to determine the depth map via localized triangular interpolation and then the reciprocal of the depth map is calculated. The generated four corner disparity maps, shown in figure 5.4 are adapted to JPEG Pleno syntax and stored in PGM format. Configuration files for the HaToy dataset are generated to test our assets in the Pleno VM. The view prediction and reconstruction results are very promising, and the Pleno community will host the HaToy assets as a light field test dataset in the JPEG image database.

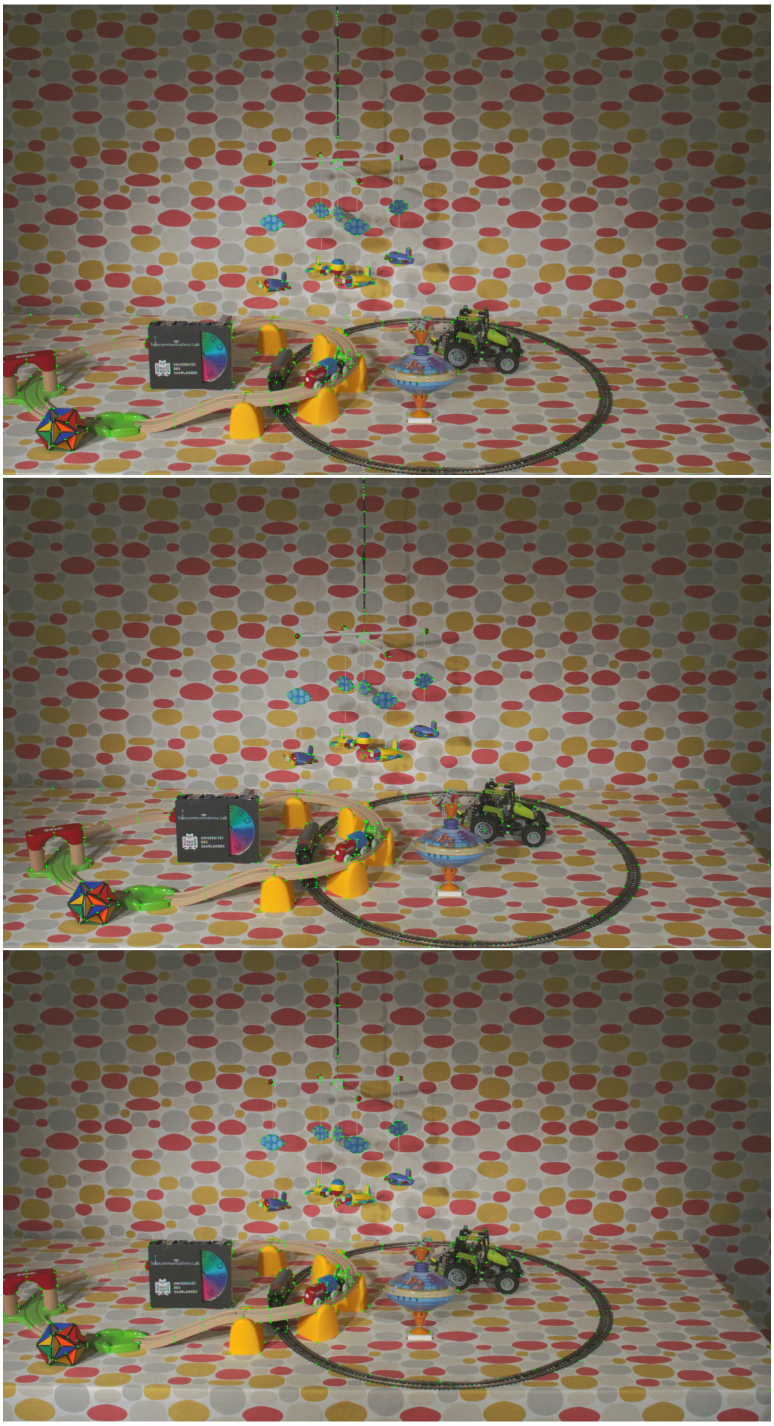


FIGURE 5.3: Adjacent camera image pairs annotated at characteristic points for the target frame 007 (top), using the adjacent frames 006 (middle), 015 (bottom) respectively (zoom in to view the marked characteristic points)

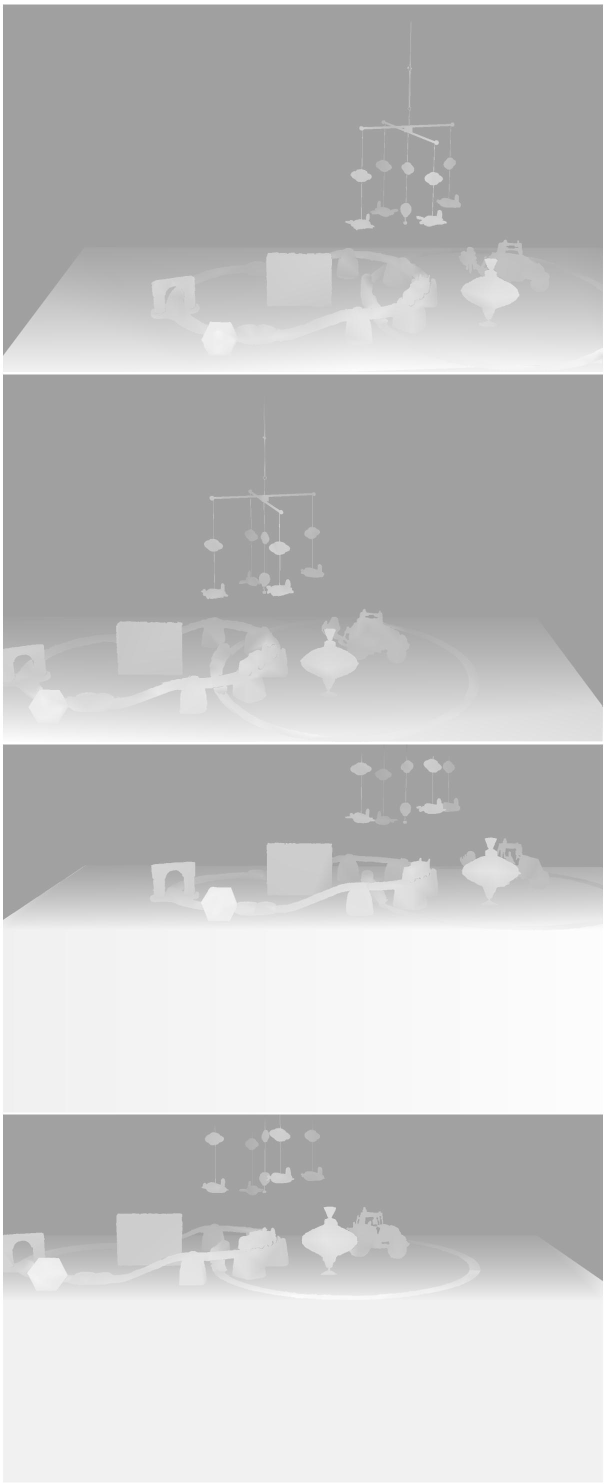


FIGURE 5.4: HaToy dataset disparity maps for four corner views - 000, 007, 056 & 063; from top to bottom respectively

5.2 MPEG-I

MPEG-I is the new era under-development standard for augmented and virtual reality applications, where "I" attributes to the "Immersive" features. The standard aims in providing natural and realistic immersive experiences for both the eyes and the ears. For instance, envision the possibility of moving freely in a concert hall during a music session or the opportunity to virtually walk around in a stadium while a sport event is happening. With MPEG-I, all this can be realised at six degrees of freedom (6DoF), either by using head mounted gears for 360° video, light field displays or free navigation in three-dimensional space. MPEG-I supports two types of coding approaches, MultiView + Depth (MVD) video coding for three-dimensional film, production and Point Cloud Coding (PCC) for three-dimensional graphic and gaming production.

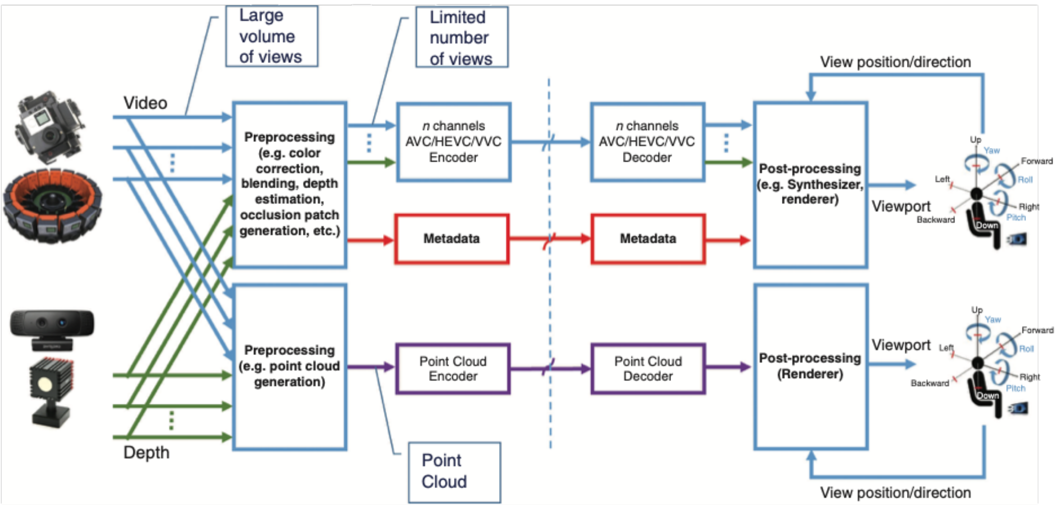

Large volume of views
Limited number of views
Video
Depth
Preprocessing (e.g. color correction, blending, depth estimation, occlusion patch generation, etc.)
Preprocessing (e.g. point cloud generation)
n channels AVC/HEVC/VVC Encoder
n channels AVC/HEVC/VVC Decoder
Metadata
Metadata
Point Cloud Encoder
Point Cloud Decoder
Point Cloud
Post-processing (e.g. Synthesizer, renderer)
Post-processing (Renderer)
View position/direction
View position/direction
Viewport
Viewport
Up
Yaw
Forward
Roll
Left
Right
Pitch
Backward
Down

FIGURE 5.5: MPEG-I generic coding and processing pipeline
Figure illustrated from [Laf+19]

The generic coding and processing pipeline of MPEG-I standard used in an immersive application is shown in figure 5.5. The pipeline illustrates the integration of both graphic and video based approaches. Depending on the type of the input data, either multi-camera array views with the associated depth information or point clouds with color intensity values and related depth information, the coding scheme is chosen. As an initial step, the input data are pre-processed for distortion removal, color correction, depth estimation, rectification and point cloud extraction. The pre-processed input data is then compressed using the state-of-the-art video compression approaches such as, AVC, HEVC, VVC or point cloud encoder. The accompanying metadata which is essential for view interpolation and reconstruction are as well transmitted with the compressed bit streams. On the decoder side, the compressed bit streams are unpacked, decoded and extracted in the graphics and video based representation formats. As a final step, the renderer performs several post-processing approaches to render an augmented or virtual reality supported data sequence which is suitable for light field displays or head mounted devices. Unlike the conventional two-dimensional video codecs, in MPEG-I video codec, the images extracted from the decoded bit streams are interpolated using a view synthesis approach creating virtual views, engaging the users in a 6DoF immersive experience. On the other hand, with the MPEG-I graphics codec, the information extracted from the decoded bit streams are used to create a point cloud of colored points in a three-dimensional space. These points are then projected onto the display using a standard OpenGL pipeline used for three-dimensional graphics. Due to several missing points that create gaps during the rendering process, the point clouds are amplified, for instance with splatting [Yif+19] approaches.

Chapter 6

Conclusion and Outlook

Plenoptics and light fields are a revived and evolving research topic. Because of the technical constraints of its acquisition, as well as the complexity and trade-offs it presents, several challenges must be resolved before light fields can be widely adopted into traditional imaging and video processing pipelines.

This thesis work presents, discusses and explores multiple approaches that facilitates the use of light field data and its capabilities in current applications. Although, several open problems have been answered, there are much more stimulating challenges to be dealt with in the future. In this chapter the formerly discussed research findings are summarised and concluded, and some of the interesting and potential future opportunities are presented.

We first focused on the basis for understanding computational imaging and its processing. The novel ideas and the different capturing devices for light fields showcased the versatility in data acquisition and the need for different tools to process the RAW data. It was made evident that light field technology is progressing rapidly and as well offers several post-processing opportunities which are welcomed in the current media community. Owing to that fact, in chapter 3 we targeted on the problem of adapting the light field data to be handled by advanced imaging file formats. PSD and OpenEXR, allows asset re-usability and also interoperability between devices and imaging applications. We developed techniques to extract and transcode high resolution light field data to professional file formats. Additionally, both the image data and metadata can be consolidated within a single container to ensure the properties of light fields be derived backwards compatible. The proposed file format conversion schemes on light light data were analyzed with post-processing approaches. Results exhibit the possibility of utilising the conversion formats without losing the underlying properties of light fields. We also laid focus on the coding performance of the different lossy and lossless compression methods and its effect on the post-process performance.

In chapter 4, we proposed techniques to optimise light field data to be compatible with state-of-the-art image and video codecs. As light field data contains multiple viewpoints captured within a restricted viewing angle, the views are highly correlated exhibiting strong redundancies. Our techniques were optimised to utilise view redundancies to achieve a better compression ratio at an optimal coding cost. The first approach was based on low-complexity re-ordering of data by pseudo-temporally sequencing the frames. In comparison to the widely used re-ordering techniques like zigzag, tiling or linewise, the proposed circular scanning outperformed consistently, as it maximised the redundancy between the frames. Experimental results showed increase in rate-distortion performance in terms of PSNR and SSIM at different bit rates for several light fields using HEVC predictive coding compared to still image codecs JPEG, JPEG 2000 and HEVC intra. We then presented a second approach where the light field views are adaptively Gaussian filtered based on superpixel segmentation and just noticeable difference threshold and then pseudo-temporally sequenced. The outcome showcased significant average bit rate reductions with almost negligible loss in the visual quality for both all-in focus and refocused light fields.

In our next approach, we introduced a predictive coding scheme for 5D light field data. Our proximity maximizer implementation generates an optimised re-ordering layout based on the camera array setup and the user desired start position. The reference lists are as well customized per frame based on the re-ordering layout to maximize the prediction gain. The results showcased that by pseudo-video re-ordering using the proposed mechanism organizes the 5D light field data in an optimal sequence exploiting the redundancies between SAIs, resulting in higher coding efficiency compared to our own previously proposed re-ordering ideas and also in comparison to coding using advanced still image codecs such as, HEIC. Overall, the core essence of all these proposed data pre-processing approaches is that, they do not introduce any changes to the codecs. Hence, the techniques can be applied to all standard video codecs and seamlessly integrated into the present and future storage and transmission services for both professional and consumer needs.

We then focused our attention towards the light field standards, JPEG Pleno and MPEG-I. As members of the JPEG committee and based on the interests from the Pleno sub-group, we designed and captured the HaToy dataset, using our 5D light field camera array. Following which, we adapted the RAW data to generate texture and disparity maps compatible to the Pleno syntax, as discussed in chapter 5. Our user-assisted semi-automatic implementation for disparity map generation produces an accurate construction of the disparity maps in comparison to the state-of-the-art depth and disparity map generation algorithms.

A prospective future work in representation of light field data will be the development of a new file format that standardises light field content. The file format needs to be robust to handle light fields captured or generated using the different data acquisition techniques. Indeed, the standard should facilitate backward compatibility feature, so the light field data can be easily adapted to traditional image and video processing pipelines. In terms of optimised reference list generation, it is still possible to achieve a better compression ratio by enlarging the group of pictures list to include sub-aperture images from consecutive frames and enabling bi-directional prediction.

As for the depth and disparity map estimation algorithms for light field data, it is still a challenging topic that requires further investigation. In addition to the commonly addressed issues such as reflectivity, transparency, occlusion and specular surfaces, light field data captured using handheld devices presents a narrow baseline with small perspective changes between views. This reduces the overall disparity range, for example, in Lytro first generation cameras (discussed in section 2.2.1.1) the disparity range is between -1 to 1 pixels. In the case of light field images captured using multi-camera array, apart from the above described issues, the implementations suffer from complex parameter selection as different views of the scene can be used as reference images. Another important aspect the available algorithms lack, is the possibility to choose the best patches/blocks from different neighbouring views to build the complete disparity map of a given image. Additionally, the complexity of the scene geometry as well adds another degree of challenge. Overall, as depth and disparity maps are an important element that is broadly used for decreasing the volume of light field texture that is coded, stored and transmitted in the processing pipeline, it is essential to develop robust algorithms to overcome inaccuracies caused by baselines, scene geometry and parameter selection for combining the best suited blocks. Furthermore, conducting research for finding solutions based on deep learning approaches could enhance the process.

To conclude, we believe that the proposed solutions in this thesis work will certainly facilitate the use of light fields in image and video processing pipelines. Moreover, it is significant to report that standardizing light field data representation and coding solutions are in the prime stage of development and we can soon expect further advancements in this domain of research.

Authored Publications

[P1] Christopher Haccius, Harini Priyadarshini Hariharan, Thorsten Herfet, Jörn Jachalsky, Wolfram Putzke-Röming, and Thomas Hach. "Infrared-Aided Superpixel Segmentation". In: *International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM)*. Phoenix, 2015.

[P2] Harini Priyadarshini Hariharan. "Infrared-Aided Superpixel Segmentation". Master Thesis. 2015: Saarland University.

[P3] Harini Priyadarshini Hariharan and Thorsten Herfet. "5D Light Field Predictive Coding". In: *Proceedings of the Networked and Electronic Media Summit (NEM)*. 2020.

[P4] Harini Priyadarshini Hariharan and Thorsten Herfet. "An analysis of preprocessed light field image compression with standard codecs". In: *Networked and Electronic Media. (NEM) Summit*. Madrid, Spain, 2017.

[P5] Harini Priyadarshini Hariharan and Thorsten Herfet. "Light Field Compression by Superpixel Based Filtering and Pseudo-Temporal Reordering". In: *IEEE International Conference on Consumer Electronics (ICCE)*. Las Vegas, 2018. DOI: 10.1109/ICCE.2018.8326153.

[P6] Harini Priyadarshini Hariharan and Thorsten Herfet. "On the implication of light field compression on post-processing algorithms". In: *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. South Korea, 2019. DOI: 10.1109/BMSB47279.2019.8971923.

[P7] Harini Priyadarshini Hariharan and Thorsten Herfet. "Optimized Predictive Coding of 5D Light Fields". In: *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. 2020. DOI: 10.1109/BMSB49480.2020.9379917.

[P8] Harini Priyadarshini Hariharan, Tobias Lange, and Thorsten Herfet. "Low complexity light field compression based on pseudo-temporal circular sequencing". In: *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, BMSB*. Best Paper Award. 2017. ISBN: 9781509049370. DOI: 10.1109/BMSB.2017.7986144.

[P9] Thorsten Herfet, Tobias Lange, and Harini Priyadarshini Hariharan. "Enabling multiview- and light field-video for veridical visual experiences". In: *2018 IEEE 4th International Conference on Computer and Communications, ICCC*. 2018. ISBN: 9781538683392. DOI: 10.1109/CompComm.2018.8780991.

[P10] Daniel Pohl, Daniel Jungmann, Bartosz Taudul, Richard Membarth, Harini Hariharan, Thorsten Herfet, and Oliver Grau. “The Next Generation of In-home Streaming: Light Fields, 5K, 10 GbE, and Foveated Compression”. In: *Proceedings of the 10th International Symposium on Multimedia Applications and Processing (MMAP)*. Best Paper Award. IEEE. Prague, Czech Republic, 2017, pp. 663–667. DOI: 10.15439/2017F16.

Bibliography

[Ach+10] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, P Fua, and S Susstrunk. "SLIC Superpixels". In: *EPFL Technical Report 149300*. 2010. ISBN: 0162-8828. DOI: 10.1109/TPAMI.2012.120.

[AW92] Edward H. Adelson and John Y.A. Wang. "Single Lens Stereo with a Plenoptic Camera". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1992). ISSN: 01628828. DOI: 10.1109/34.121783.

[AS17] Martin Alain and Aljosa Smolic. "Light field denoising by sparse 5D transform domain collaborative filtering". In: *2017 IEEE 19th International Workshop on Multimedia Signal Processing, MMSP 2017*. 2017. ISBN: 9781509036493. DOI: 10.1109/MMSP.2017.8122232.

[AS18] Martin Alain and Aljosa Smolic. "Light Field Super-Resolution via LFBM5D Sparse Coding". In: *Proceedings - International Conference on Image Processing, ICIP*. 2018. ISBN: 9781479970612. DOI: 10.1109/ICIP.2018.8451162.

[Ast+20] Pekka Astola, Luis A. da Silva Cruz, Eduardo A. B. da Silva, Touradj Ebrahimi, Pedro Garcia Freitas, Antonin Gilles, Kwan-Jung Oh, Carla Pagliari, Fernando Pereira, Cristian Perra, Stuart Perry, Antonio M. G. Pinheiro, Peter Schelkens, Ismael Seidel, and Ioan Tabus. "JPEG Pleno: Standardizing a Coding Framework and Tools for Plenoptic Imaging Modalities". In: *ITU Journal: ICT Discoveries* 3.1 (2020).

[AT18] Pekka Astola and Ioan Tabus. "Light field compression of HDCA images combining linear prediction and JPEG 2000". In: *European Signal Processing Conference*. Vol. 2018-September. 2018. DOI: 10.23919/EUSIPCO.2018.8553482.

[AT19a] Pekka Astola and Ioan Tabus. "Coding of Light Fields Using Disparity-Based Sparse Prediction". In: *IEEE Access* 7 (2019). ISSN: 21693536. DOI: 10.1109/ACCESS.2019.2957934.

[AT19b] Pekka Astola and Ioan Tabus. "WaSP: Hierarchical Warping, Merging, and Sparse Prediction for Light Field Image Compression". In: *Proceedings - European Workshop on Visual Information Processing, EUVIP*. Vol. 2018-November. 2019. DOI: 10.1109/EUVIP.2018.8611756.

[BBM87] Robert C. Bolles, H. Harlyn Baker, and David H. Marimont. "Epipolar-plane image analysis: An approach to determining structure from motion". In: *International Journal of Computer Vision* (1987). ISSN: 09205691. DOI: 10.1007/BF00128525.

[Cha+04] Shing Chow Chan, King To Ng, Zhi Feng Gan, Kin Lok Chan, and Heung Yeung Shum. "The plenoptic videos: Capturing, rendering and compression". In: *Proceedings - IEEE International Symposium on Circuits and Systems*. 2004. DOI: 10.1109/iscas.2004.1328894.

[CAS20] Yang Chen, Martin Alain, and Aljosa Smolic. *Fast and Accurate Optical Flow based Depth Map Estimation from Light Fields*. 2020.

[CL95] Chun Hsien Chou and Yun Chin Li. "A Perceptually Tuned Subband Image Coder Based on the Measure of Just-Noticeable-Distortion Profile". In: *IEEE Transactions on Circuits and Systems for Video Technology* (1995). ISSN: 15582205. DOI: 10.1109/76.475889.

[CM02] Dorin Comaniciu and Peter Meer. "Mean shift: A robust approach toward feature space analysis". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2002). ISSN: 01628828. DOI: 10.1109/34.1000236.

[Con+18] Ruhan Conceicao, Marcelo Porto, Bruno Zatt, and Luciano Agostini. "LF-CAE: Context-Adaptive Encoding for Lenslet Light Fields Using HEVC". In: *Proceedings - International Conference on Image Processing, ICIP*. 2018. ISBN: 9781479970612. DOI: 10.1109/ICIP.2018.8451345.

[Dan] Donald Dansereau. *Light Field Toolbox for MATLAB*. URL: https://dgd.vision/Tools/LFToolbox/ (visited on 01/12/2021).

[DPW13] Donald Dansereau, Oscar Pizarro, and Stefan B. Williams. "Decoding, calibration and rectification for lenselet-based plenoptic cameras". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2013. DOI: 10.1109/CVPR.2013.137.

[DBM19] Walid Darwish, Quentin Bolsee, and Adrian Munteanu. "Plenoptic Camera Calibration Based on Sub-Aperture Images". In: *International Conference on Image Processing, ICIP (2019)*. 2019, pp. 3527–3531. DOI: 10.1109/ICIP.2019.8803473.

[de +20] Gustavo de Oliveira Alves, Murilo Bresciani de Carvalho, Carla L. Pagliari, Pedro Garcia Freitas, Ismael Seidel, Marcio Pinto Pereira, Carla Florentino Schueler Vieira, Vanessa Testoni, Fernando Pereira, and Eduardo A.B. da Silva. "The JPEG pleno light field coding standard 4D-transform mode: How to design an efficient 4D-native codec". In: *IEEE Access* 8 (2020). ISSN: 21693536. DOI: 10.1109/ACCESS.2020.3024844.

[Din+15] Lei Ding, Ge Li, Ronggang Wang, and Wenmin Wang. "Video pre-processing with JND-based Gaussian filtering of superpixels". In: *Visual Information Processing and Communication VI*. 2015. ISBN: 9781628415001. DOI: 10.1117/12.2083818.

[Dom+17] Marek Domański, Olgierd Stankiewicz, Krzysztof Wegner, and Tomasz Grajek. "Immersive visual media - MPEG-I: 360 video, virtual navigation and beyond". In: *International Conference on Systems, Signals, and Image Processing*. 2017. DOI: 10.1109/IWSSIP.2017.7965623.

[Ebr+16] Touradj Ebrahimi, Siegfried Foessel, Fernando Pereira, and Peter Schelkens. "JPEG Pleno: Toward an Efficient Representation of Visual Reality". In: *IEEE Multimedia* (2016). ISSN: 19410166. DOI: 10.1109/MMUL.2016.64.

[Far46] Michael Faraday. "Thoughts on ray-vibrations". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* (1846). ISSN: 1941-5966. DOI: 10.1080/14786444608645431.

[FK05] Ulrich Fecker and André Kaup. "H.264/AVC-compatible coding of dynamic light fields using transposed picture ordering". In: *13th European Signal Processing Conference, EUSIPCO 2005*. 2005. ISBN: 1604238216.

[FH04] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. "Efficient graph-based image segmentation". In: *International Journal of Computer Vision* (2004). ISSN: 09205691. DOI: 10.1023/B:VISI.0000022288.19776.77.

[Ger36] Arun Gershun. "The Light Field". In: *Journal of Mathematics and Physics* (1936). DOI: 10.1002/sapm193918151.

[HLC19] Thorsten Herfet, Tobias Lange, and Kelvin Chelli. "5D Light Field Video Capture". In: *The 16th ACM SIGGRAPH European Conference on Visual Media Production*. BFI Southbank, London, UK, 2019.

[Hon+17] Katrin Honauer, Ole Johannsen, Daniel Kondermann, and Bastian Goldluecke. "A dataset and evaluation methodology for depth estimation on 4D light fields". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 10113 LNCS. 2017. DOI: 10.1007/978-3-319-54187-7_2.

[Hua+15] Chao Tsung Huang, Jui Chin, Hong Hui Chen, Yu Wen Wang, and Liang Gee Chen. "Fast realistic refocusing for sparse light fields". In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. 2015. ISBN: 9781467369978. DOI: 10.1109/ICASSP.2015.7178155.

[Ima+19] Kota Imaeda, Kohei Isechi, Keita Takahashi, Toshiaki Fujii, Yukihiro Bandoh, Takehito Miyazawa, Seishi Takamura, and Atsushi Shimizu. “LF-TSP: Traveling salesman problem for HEVC-based light-field coding”. In: *2019 IEEE International Conference on Visual Communications and Image Processing, VCIP 2019*. 2019. ISBN: 9781728137230. DOI: 10.1109/VCIP47243.2019.8965837.

[Int88] International Telecommunication Union. “Information technology - Digital compression and coding of continuous - tone still images - requirements and guidelines”. In: *Study Group VIII, ITU, Geneva* (1988).

[ITU02] ITU. “Information technology: JPEG 2000 image coding system: Core coding system”. In: *International Telecommunications Union* (2002).

[Joh+13] Ole Johannsen, Christian Heinze, Bastian Goldluecke, and Christian Perwaß. “On the calibration of focused plenoptic cameras”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2013. ISBN: 9783642449635. DOI: 10.1007/978-3-642-44964-2_15.

[Kai13] Florian Kainz. *OpenEXR File Layout*. 2013. URL: https://www.openexr.com/documentation/openexrfilelayout.pdf (visited on 01/28/2021).

[Kle] *K | Lens*. 2020. URL: https://www.k-lens.de (visited on 10/30/2020).

[Kno19] Thomas Knoll. *Adobe Photoshop File Formats Specification*. 2019. (Visited on 01/27/2021).

[KKG14] Amit Kumar, Matthew A. Killingsworth, and Thomas Gilovich. “Waiting for Merlot: Anticipatory Consumption of Experiential and Material Purchases”. In: *Psychological Science* 25.10 (2014). ISSN: 14679280. DOI: 10.1177/0956797614546556.

[Kwa+03] Vivek Kwatra, Arno Schödl, Irfan Essa, Greg Turk, and Aaron Bobick. “Graphcut textures: Image and video synthesis using graph cuts”. In: *ACM SIGGRAPH 2003 Papers, SIGGRAPH '03*. 2003. ISBN: 1581137095. DOI: 10.1145/1201775.882264.

[Laf+19] Gauthier Lafruit, Daniele Bonatto, Christian Tulvan, Marius Preda, and Lu Yu. “Understanding MPEG-I coding standardization in immersive VR/AR applications”. In: *SMPTE Motion Imaging Journal*. Vol. 128. 10. 2019. DOI: 10.5594/JMI.2019.2941362.

[Lev+09] Alex Levinshtein, Adrian Stere, Kiriakos N. Kutulakos, David J. Fleet, Sven J. Dickinson, and Kaleem Siddiqi. “TurboPixels: Fast superpixels using geometric flows”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2009. DOI: 10.1109/TPAMI.2009.96.

[LH96] Marc Levoy and Pat Hanrahan. “Light field rendering”. In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996*. 1996. ISBN: 0897917464. DOI: 10.1145/237170.237199.

[Lip08] Gabriel Lippman. “Epreuves reversibles donnant la sensation du relief”. In: *J. Phys. Theor. Appl.* 7 (1908).

[Liu+16] Dong Liu, Lizhi Wang, Li Li, Xiong Zhiwei, Feng Wu, and Zeng Wenjun. “Pseudo-sequence-based light field image compression”. In: *2016 IEEE International Conference on Multimedia and Expo Workshop, ICMEW 2016*. 2016. ISBN: 9781509015528. DOI: 10.1109/ICMEW.2016.7574674.

[LG09] Andrew Lumsdaine and Todor Georgiev. “The focused plenoptic camera”. In: *2009 IEEE International Conference on Computational Photography, ICCP 09*. 2009. ISBN: 9781424445332. DOI: 10.1109/ICCPHOT.2009.5559008.

[Lyi] *Lytro Illum - Professional Light Field Camera*. 2014. URL: http://lightfield-forum.com/lytro/lytro-illum-professional-light-field-camera/ (visited on 10/30/2020).

[Lyc] *Lytro Light Field Cameras*. 2012. URL: http://lightfield-forum.com/lytro/lytro-lightfield-camera/ (visited on 10/30/2020).

[MG00] Marcus Magnor and Bernd Girod. “Data compression for light-field rendering”. In: *IEEE Transactions on Circuits and Systems for Video Technology* (2000). ISSN: 10518215. DOI: 10.1109/76.836278.

[Mon+19] Ricardo J.S. Monteiro, Nuno M.M. Rodrigues, Sérgio M.M. Faria, and Paulo J.L. Nunes. “Optimized reference picture selection for light field image coding”. In: *European Signal Processing Conference*. 2019. ISBN: 9789082797039. DOI: 10.23919/EUSIPCO.2019.8902555.

[MS81] Parry. Moon and Domina Eberle. Spencer. “The photic field”. In: *Cambridge* (1981).

[Moo+08] Alastair P. Moore, Simon J.D. Prince, Jonathan Warrell, Umar Mohammed, and Graham Jones. “Superpixel lattices”. In: *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*. 2008. ISBN: 9781424422432. DOI: 10.1109/CVPR.2008.4587471.

[Ng+05] Ren Ng, Marc Levoy, Mathieu Bredif, Gene Duval, Mark Horowitz, and Pat Hanrahan. “Light Field Photography with a Hand-held Plenoptic Camera”. 2005.

[Ove+18] Ryan S. Overbeck, Daniel Erickson, Daniel Evangelakos, Matt Pharr, and Paul Debevec. “A system for acquiring, processing, and rendering panoramic light field stills for virtual reality”. In: *SIGGRAPH Asia 2018 Technical Papers, SIGGRAPH Asia 2018*. 2018. ISBN: 9781450360081. DOI: 10.1145/3272127.3275031. arXiv: 1810.08860.

[PKV18] Luca Palmieri, Reinhard Koch, and Ron Op Het Veld. "The Plenoptic 2.0 Toolbox: Benchmarking of Depth Estimation Methods for MLA-Based Focused Plenoptic Cameras". In: *Proceedings - International Conference on Image Processing, ICIP*. 2018. ISBN: 9781479970612. DOI: 10.1109/ICIP.2018.8451073.

[Per15] Cristian Perra. "Lossless plenoptic image compression using adaptive block differential prediction". In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. 2015. ISBN: 9781467369978. DOI: 10.1109/ICASSP.2015.7178166.

[PA16] Cristian. Perra and Pedro Assuncao. "High efficiency coding of light field images based on tiling and pseudo-temporal data arrangement". In: *2016 IEEE International Conference on Multimedia and Expo Workshop, ICMEW 2016*. 2016. ISBN: 9781509015528. DOI: 10.1109/ICMEW.2016.7574671.

[Per+19] Cristian Perra, Pekka Astola, Eduardo A. B. da Silva, Hesam Khanmohammad, Carla Pagliari, Peter Schelkens, and Ioan Tabus. "Performance analysis of JPEG Pleno light field coding". In: 2019. DOI: 10.1117/12.2528391.

[PG17] Cristian Perra and Daniele Giusto. "JPEG 2000 compression of unfocused light field images based on lenslet array slicing". In: *2017 IEEE International Conference on Consumer Electronics, ICCE 2017*. 2017. ISBN: 9781509055449. DOI: 10.1109/ICCE.2017.7889217.

[Ray] *Raytrix*. 2019. URL: https://raytrix.de (visited on 10/30/2020).

[RE16] Martin Rerabek and Touradj Ebrahimi. "New Light Field Image Dataset". In: *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*. 2016.

[RAE16] Thomas Richter, Alessandro Artusi, and Touradj Ebrahimi. "JPEG XT: A New Family of JPEG Backward-Compatible Standards". In: *IEEE Multimedia* (2016). ISSN: 1070986X. DOI: 10.1109/MMUL.2016.49.

[RSM20] Ségolène Rogge, Ionut Schiopu, and Adrian Munteanu. "Depth estimation for light-field images using stereo matching and convolutional neural networks". In: *Sensors (Switzerland)* 20.21 (2020). ISSN: 14248220. DOI: 10.3390/s20216188.

[Sal07] David Salomon. *Data compression: The complete reference*. 2007. ISBN: 1846286026. DOI: 10.1007/978-1-84628-603-2.

[Sch+18] Peter Schelkens, Zahir Alpaslan, Ioan Tabus, Touradj Ebrahimi, Kwan-Jung Oh, Antonio M. G. Pinheiro, Zhibo Chen, and Fernando M. Pereira. "JPEG Pleno: a standard framework for representing and signaling plenoptic modalities". In: 2018. DOI: 10.1117/12.2323404.

[Sch+19] Peter Schelkens, Touradj Ebrahimi, Antonin Gilles, Patrick Gioia, Kwan Jung Oh, Fernando Pereira, Cristian Perra, and Antonio M.G. Pinheiro. "JPEG Pleno: Providing representation interoperability for holographic applications and devices". In: *ETRI Journal* 41.1 (2019). ISSN: 22337326. DOI: 10.4218/etrij.2018-0509.

[Skl88] Bernard Sklar. "Companding Characteristics". In: *Digital Communications: Fundamentals and Applications*. New Jersy: Prentice-Hall, 1988, pp. 84–85.

[Sul+12] Gary J. Sullivan, Jens Rainer Ohm, Woo Jin Han, and Thomas Wiegand. "Overview of the high efficiency video coding (HEVC) standard". In: *IEEE Transactions on Circuits and Systems for Video Technology* (2012). ISSN: 10518215. DOI: 10.1109/TCSVT.2012.2221191.

[Tag+10] Yuichi Taguchi, Amit Agrawal, Srikumar Ramalingam, and Ashok Veeraraghavan. "Axial light field for curved mirrors: Reflect your perspective, widen your view". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2010. ISBN: 9781424469840. DOI: 10.1109/CVPR.2010.5540172.

[TFT14] Chelsea M. Thomason, Tim F. Fahringer, and Brian S. Thurow. "Calibration of a microlens array for a plenoptic camera". In: *52nd Aerospace Sciences Meeting*. 2014. ISBN: 9781624102561. DOI: 10.2514/6.2014-0396.

[Tsa+17] Dorian Tsai, Donald G. Dansereau, Thierry Peynot, and Peter Corke. "Image-Based Visual Servoing with Light Field Cameras". In: *IEEE Robotics and Automation Letters* (2017). ISSN: 23773766. DOI: 10.1109/LRA.2017.2654544.

[Unf] *Unfolding 2.0*. 2019. URL: https://degas.filmakademie.de/nextcloud/s/DLDpDP3N66ZBsfk?path=%2F (visited on 01/06/2021).

[VA08a] Vaibhav Vaish and Andrew Adams. *The (New) Stanford Light Field Archive*. 2008. URL: http://lightfield.stanford.edu/lfs.html (visited on 01/06/2021).

[VA08b] Vaibhav Vaish and Andrew Adams. *The (New) Stanford Light Field Archive*. 2008. URL: http://lightfield.stanford.edu/acq.html (visited on 01/06/2021).

[VS08] Andrea Vedaldi and Stefano Soatto. "Quick shift and kernel methods for mode seeking". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2008. ISBN: 3540886923. DOI: 10.1007/978-3-540-88693-8_52.

[VBM10] Olga Veksler, Yuri Boykov, and Paria Mehrani. "Superpixels and supervoxels in an energy optimization framework". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2010. ISBN: 3642155545. DOI: 10.1007/978-3-642-15555-0_16.

[Ven+13] Kartik Venkataraman, Dan Lelescu, Jacques Duparrë, Andrew McMahon, Gabriel Molina, Priyam Chatterjee, Robert Mullism, and Shree Nayar. "PiCam: An ultra-thin high performance monolithic camera array". In: *ACM Transactions on Graphics* (2013). ISSN: 07300301. DOI: 10.1145/2508363.2508390.

[Via+17] Alessandro Vianello, Giulio Manfredi, Maximilian Diebold, and Bernd Jähne. "3D reconstruction by a combined structure tensor and Hough transform light field approach". In: *Technisches Messen* (2017). ISSN: 21967113. DOI: 10.1515/teme-2017-0010.

[Vie+15] Alexandre Vieira, Helder Duarte, Cristian Perra, Luis Tavora, and Pedro Assuncao. "Data formats for high efficiency coding of Lytro-Illum light fields". In: *5th International Conference on Image Processing, Theory, Tools and Applications 2015, IPTA 2015*. 2015. ISBN: 9781479986354. DOI: 10.1109/IPTA.2015.7367195.

[VVS91] Luc Vincent, Luc Vincent, and Pierre Soille. "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1991). ISSN: 01628828. DOI: 10.1109/34.87344.

[Wil+05] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. "High performance imaging using large camera arrays". In: *ACM Transactions on Graphics*. 2005. DOI: 10.1145/1073204.1073259.

[Yif+19] Wang Yifan, Felice Serena, Shihao Wu, Cengiz Öztireli, and Olga Sorkine-Hornung. "Differentiable surface splatting for point-based geometry processing". In: *ACM Transactions on Graphics* 38.6 (2019). ISSN: 15577368. DOI: 10.1145/3355089.3356513.

[ZC04] Cha Zhang and Tsuhan Chen. "A self-reconfigurable camera array". In: *ACM SIGGRAPH 2004 Sketches, SIGGRAPH'04*. 2004. DOI: 10.1145/1186223.1186412.

[Zha00] Zhengyou Zhang. "A flexible new technique for camera calibration". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2000). ISSN: 01628828. DOI: 10.1109/34.888718.