Advanced Editing Methods for Image and Video Sequences

Miguel A. Granados Velásquez

Thesis for obtaining the title of
Doctor of Engineering Science (Dr.-Ing.)
of the Faculty of Natural Sciences and Technology I
of Saarland University

Filed on February 19st, 2013
Saarbrücken, Germany

UNIVERSITÄT
DES
SAARLANDES

max planck institut
informatik

Dekan – Dean
Prof. Dr. Mark Groves Universität des Saarlandes Saarbrücken, Germany

Betreuender Hochschullehrer – Supervisor
Prof. Dr. Christian Theobalt MPI Informatik Saarbrücken, Germany

Gutachter – Reviewers

Prof. Dr. Christian Theobalt	MPI Informatik	Saarbrücken, Germany
Prof. Dr. Hans-Peter Seidel	MPI Informatik	Saarbrücken, Germany
Prof. Dr. Jan Kautz	University College London	London, United Kingdom

Kolloquiums – Defense

Datum – Date
September 10, 2013, in Saarbrücken
Vorsitzender – Head of Colloquium:

Prof. Dr. Philipp Slusallek	Universität des Saarlandes	Saarbrücken, Germany

Prüfer – Examiners:

Prof. Dr. Christian Theobalt	MPI Informatik	Saarbrücken, Germany
Prof. Dr. Hans-Peter Seidel	MPI Informatik	Saarbrücken, Germany

Protokoll – Reporter:

Dr. Kwang In Kim	MPI Informatik	Saarbrücken, Germany

Miguel A. Granados Velásquez
Max-Planck-Institut für Informatik
Campus E14
D-66123, Saarbrücken
granados@mpii.de

To my journey companions

Abstract

In the context of image and video editing, this thesis proposes methods for modifying the semantic content of a recorded scene. Two different editing problems are approached: First, the removal of ghosting artifacts from high dynamic range (HDR) images recovered from exposure sequences, and second, the removal of objects from video sequences recorded with and without camera motion. These editings need to be performed in a way that the result looks plausible to humans, but without having to recover detailed models about the content of the scene, e.g. its geometry, reflectance, or illumination.

The proposed editing methods add new key ingredients, such as camera noise models and global optimization frameworks, that help achieving results that surpass the capabilities of state-of-the-art methods. Using these ingredients, each proposed method defines local visual properties that approximate well the specific editing requirements of each task. These properties are then encoded into a energy function that, when globally minimized, produces the required editing results. The optimization of such energy functions corresponds to Bayesian inference problems that are solved efficiently using graph cuts.

The proposed methods are demonstrated to outperform other state-of-the-art methods. Furthermore, they are demonstrated to work well on complex real-world scenarios that have not been previously addressed in the literature, i.e., highly cluttered scenes for HDR deghosting, and highly dynamic scenes and unconstraint camera motion for object removal from videos.

Kurzfassung

Diese Arbeit schlägt Methoden zur Änderung des semantischen Inhalts einer aufgenommenen Szene im Kontext der Bild-und Videobearbeitung vor. Zwei unterschiedliche Bearbeitungsmethoden werden angesprochen: Erstens, das Entfernen von Ghosting Artifacts (Geist-ähnliche Artefakte) aus High Dynamic Range (HDR) Bildern welche von Belichtungsreihen erstellt wurden und zweitens, das Entfernen von Objekten aus Videosequenzen mit und ohne Kamerabewegung. Das Bearbeiten muss in einer Weise durchgeführt werden, dass das Ergebnis für den Menschen plausibel aussieht, aber ohne das detaillierte Modelle des Szeneninhalts rekonstruiert werden müssen, z.B. die Geometrie, das Reflexionsverhalten, oder Beleuchtungseigenschaften.

Die vorgeschlagenen Bearbeitungsmethoden beinhalten neuartige Elemente, etwa Kameralärm-Modelle und globale Optimierungs-Systeme, mit deren Hilfe es möglich ist die Eigenschaften der modernsten existierenden Methoden zu übertreffen. Mit Hilfe dieser Elemente definieren die vorgeschlagenen Methoden lokale visuelle Eigenschaften welche die beschriebenen Bearbeitungsmethoden gut annähern. Diese Eigenschaften werden dann als Energiefunktion codiert, welche, nach globalem minimieren, die gewünschten Bearbeitung liefert. Die Optimierung solcher Energiefunktionen entspricht dem Bayes'schen Inferenz Modell welches effizient mittels Graph-Cut Algorithmen gelöst werden kann.

Es wird gezeigt, dass die vorgeschlagenen Methoden den heutigen Stand der Technik übertreffen. Darüber hinaus sind sie nachweislich gut auf komplexe natürliche Szenarien anwendbar, welche in der existierenden Literatur bisher noch nicht angegangen wurden, d.h. sehr unübersichtliche Szenen für HDR Deghosting und sehr dynamische Szenen und unbeschränkte Kamerabewegungen für das Entfernen von Objekten aus Videosequenzen.

Summary

This thesis proposes methods for editing the semantic content of video and image sequences but without requiring a semantic understanding of the scene content. Two different editing problems are approached: First, the removal of ghosting artifacts from high dynamic range (HDR) images that are reconstructed from exposure sequences (i.e., sequences where every image has a different exposure time). Second, the removal of unwanted objects from video sequences that are recorded with and without camera motion. The fundamental requirement of these editing operations is that they need to be performed in a way that the result looks plausible to humans, but without having to construct complex models of the scene content, such as models for the shape and motion, the reflectance of surfaces, or the light sources.

The first editing problem is to reconstruct ghost-free HDR images of a highly dynamic scene by averaging the images in a given exposure sequence. For this purpose, a camera model is used first to predict the noise distribution of the input images. This distribution is then used to detect objects that moved between images so that only sets of consistent images are included in the average. In this way, ghosting artifacts are prevented from appearing in the final HDR image. Additionally, the same noise model is exploited for improving the quality of other tasks related to HDR image processing, including HDR image denoising, and noise-optimal HDR reconstruction.

The second task is to remove objects from video sequences by inpainting or completing the part of the scene that they occluded. The inpainting is performed by reusing other suitable instances of the occluded scene that might be available in the video, even in situations where the occluded content is dynamic. This strategy exploits the high degree of visual redundancy generally found in video sequences. For this task, two methods are proposed: First, a method that inpaints dynamic objects observed with static cameras, and second, a method that inpaints static objects observed with moving cameras.

The proposed editing methods add new key ingredients, such as camera noise models and global optimization frameworks, that help achieving re-

sults that surpass the capabilities of state-of-the-art methods. Each editing method is defined in two steps: First, it defines local visual properties that are a good approximation of the particular editing requirements and of the general requirement of producing plausible results. Second, these properties are encoded into a energy functional that, when globally minimized, produces the desired editing results. The optimization of such energy functions corresponds to Bayesian inference problems, which can be efficiently solved using graph cuts.

The proposed methods are experimentally demonstrated to outperform other state-of-the-art methods in terms of the quality and plausibility of the resulting editings. Furthermore, the proposed methods are demonstrated to work well on complex real-world scenarios that have not been previously addressed in the literature. These scenarios include highly cluttered scenes in the context of HDR deghosting, and highly dynamic scenes and unconstraint camera motion in the context of video inpainting.

Zusammenfassung

Diese Arbeit schlägt Methoden für die Bearbeitung des semantischen Inhalts von Video-und Bildsequenzen vor, ohne eine semantisches Verständnis des Szeneninhalts zu erfordern. Zwei unterschiedliche Bearbeitungsmöglichkeiten werden angesprochen: Erstens, die Entfernung von Ghosting Artifacts aus High Dynamic Range (HDR) Bildern, welche von Belichtungsreihen erstellt wurden (d.h. Sequenzen bei denen jedes Bild eine andere Belichtungszeit hat). Zweitens die Entfernung von unerwünschten Objekten aus Videosequenzen, die mit oder ohne Kamerabewegung aufgezeichnet wurden. Die Grundvoraussetzung dieser Bearbeitungsvorgänge ist, dass das Ergebnis für den Menschen plausibel aussieht, aber ohne das detaillierte Modelle des Szeneninhalts rekonstruiert werden müssen, z.B. die Form und Bewegung, das Reflexionsverhalten von Oberflächen, oder Lichtquellen Eigenschaften.

Das Ziel der ersten Bearbeitungsmethode ist es HDR-Bilder ohne Ghosting Artefakte von einer hochdynamischen Szene durch Mittlung der Bilder einer Belichtungsreihe zu rekonstruieren. Zu diesem Zweck wird ein Kameramodell verwendet welches die Verteilung des Rauschens des Eingabebildes vorhersagt. Diese Verteilung wird dann verwendet, um Objekte zu erkennen welche sich zwischen den Aufnahmen bewegt haben, so dass nur Gruppen mit konsistenten Bildern für die Durchschnittsbildung verwendet werden. Auf diese Weise wird das Auftreten von Ghosting Artefakte im endgültigen HDR-Bild vermieden. Darüber hinaus wird das gleiche Rauschmodell zur Verbesserung der Qualität von anderen Aspekten der HDR Bildbearbeitung verwendet, darunter HDR-Bild Rauschunterdrückung und Rauschoptimale HDR Rekonstruktion.

Der zweite Schwerpunkt ist, Objekte aus Videosequenzen durch Inpainting und die Vervollständigung der verdeckten Szenenteile zu entfernen. Das Inpainting wird durch Wiederverwendung geeigneter Instanzen der verdeckten Szene welche möglicherweise an anderer Stelle in dem Video vorhanden sind erreicht, auch in Situationen, in denen der verdeckte Inhalt dynamisch ist. Diese Methode nutzt die in der Regel hohe visuelle Redundanz von Videosequenzen. Für diese Aufgabe werden zwei Methoden vorgeschlagen:

Erstens, eine Methode, die dynamische Objekte, welche mit einer statischen Kamera aufgenommen wurden, ersetzt. Und zweitens eine Methode die statische Objekte, welche mit einer beweglichen Kamera aufgenommen wurden, ersetzt.

Die vorgeschlagenen Bearbeitungsmethoden enthalten neuartige Elemente, wie beispielsweise das Kameralärm-Modell und das globale Optimierungssystem, welche ermöglichen die Ergebnisse von state-of-the-art Methoden zu übertreffen. Jedes Bearbeitungsverfahren wird in zwei Stufen definiert: Erstens definiert es lokale visuelle Eigenschaften welche eine gute Annäherung an die gewünschten Bearbeitungsmethoden und an die allgemeinen Anforderung für das Erreichen plausibler Ergebnisse darstellen. Zweitens werden diese Eigenschaften in Energiefunktionen kodiert, welche, wenn global minimiert, die gewünschten Bearbeitungsergebnisse liefern. Die Optimierung solcher Energiefunktionen entspricht dem Bayes'sche Inferenz Modell welches effizient mittels Graph-Cut Algorithmen gelöst werden kann.

Es wird experimentell nachgewiesen, dass die vorgeschlagenen Methoden existierende Methoden in Bezug auf die Qualität und Plausibilität der Bearbeitungsresultate übertreffen. Ferner sind die vorgeschlagenen Methoden nachweislich gut auf komplexe natürliche Szenarien anwendbar, welche in der existierenden Literatur bisher noch nicht angegangen wurden. Beispielsweise sehr unübersichtliche Szenen für HDR Deghosting und sehr dynamische Szenen und unbeschränkte Kamerabewegungen für Video Inpainting.

Acknowledgements

My most sincere gratitude to Prof. Dr. Hans-Peter Seidel for welcoming me into his group, and for providing a warm and fruitful environment for conducting this work. To my supervisor, Prof. Dr. Christian Theobalt for helping me in times of transition, for teaching me to think and aim beyond what I thought is possible, and to not give up before even trying. To my adviser Dr. Kwang In Kim for his endless patience in explaining me even the most simple things, for making sure that scientific rigor was always present, for helping me beyond duty in writing and revising manuscripts. My sincere gratitude to my co-authors Prof. Dr. Jan Kautz, Dr. James Tompkin, Dr. Michael Wand, and Boris Ajdin, for the timely discussions, for providing me with ideas, and for helping me focus on the important goals; without their help this work would have been impossible. To my former supervisor, Prof. Dr. Hendrik P.A. Lensch for encouraging me to pursue research, and welcoming me as his PhD student; to the members of his former group at MPI Informatik, Martin Fuchs, Christian Fuchs, Matthias Hullin, and Tongbo Chen for their priceless example on how top level research is performed. To my fellow researchers in the Graphics, Vision, and Video Group at MPI Informatik, especially to Nils Hasler, Levi Valgaerts, and Kiran Varanasi, for their timely and valuable feedback, and to Helge Rhodin for his kind help with translation. To my fellow researchers in the Computer Graphics Group at MPI Informatik, for making my stay always fun and interesting. To Martin Sunkel, Peter Grosche, and the people at the Service Desk for their prompt help with all matters of infrastructure. To the secretaries of the group, Sabine Budde, Conny Liegl, and Ellen Fries, for helping me with all kind of everyday problems, and for helping me cope with the life in a foreign country. To my friends in Saarbrücken, especially to Fidel Ramírez, José David Gomez, and José Brito whose brilliant minds enriched my life and made me grow in every way that matters. And most importantly, to my beloved wife Lina Ruiz, whose constant support during these years has been most invaluable, and whose company has made my life the most exciting learning experience.

Contents

CHAPTER 1

Introduction

In the last decade, several computer vision and graphical editing tasks have become mature enough to be routinely applied in industries such as publishing, advertisement, and movie and television production. For instance, currently available commercial software [Adobeb, Adobea, Microsoft, HDRSoft] includes algorithms for several high level image and video processing tasks which were previously unavailable to the general public. Such methods include image dynamic range enhancement [Debevec97], camera tracking [Pollefeys02a], image compositing [Agarwala04], image inpainting, re-targeting and reshuffling [Barnes09], rotoscoping [Bai09], and video stabilization [Liu11]. Although these methods are not perfect, they can already exempt artists (at least partly) from performing manually these time consuming tasks, so they can focus on other more advanced endeavors.

However, artists still perform other editing tasks manually. These tasks are actively researched and include problems such as video re-targeting [Hu10], video inpainting [Wexler07, Patwardhan05], and video decomposition into motion layers [Schoenemann12]. These have in common that they require high level editings that depend on the content of the scene (e.g. in video re-targeting and layer decomposition), or that modify its content (e.g. in video inpainting). These tasks are very challenging, and in general, solutions have been proposed first for still images, and subsequently, for video and image sequences. This can be explained as the additional (temporal) dimension of sequences implies an extra order of algorithmic complexity. This makes the editing of video and image sequences much more challenging than the editing of still images.

The need for performing more automatic editings becomes more evident

when the footage is recorded in uncontrolled scenarios outside of the studio, such as in crowded, public places. In such scenarios, it is often required to edit away scene elements that are not meant to be in the final composition. Such elements can include pedestrians, cars, street signs, public wiring, and advertisement, but also crew members or equipment that need to be in the scene for technical reasons. This type of editings are currently available in commercial software for the case of still images [Adobeb], but they are either not available or not mature enough to be made available to the general public for the case of video and image sequences, due to their higher complexity. For instance, in the interaction of our group with the movie industry, we learned that operations such as layer decomposition, or removing unwanted scene elements from videos are still performed by artists in a frame-by-frame basis. Therefore, if automatic methods for high level editing become faster and more reliable, the editing process can be made less time consuming, and be more widely adopted by the public.

In any case, it is very challenging to develop video and image editing methods whose goal is to change the *meaning* of the scene. The main difficulty arises from the requirement that such methods should produce videos and images that look *plausible* or semantically correct to humans, but without having an *understanding* about the semantic content of the scene. For many tasks, this type of editings have been made possible by the application of optimization methods that aim at fulfilling the editing requirements, while at the same time satisfying the plausibility constraints, despite the fact that these latter are more challenging to define precisely. In many cases, these constraints can be achieved without assuming any type of understanding or *model* of the scene, for instance, when the editing requirements can be expressed using low-level visual cues available in the locality of every pixel. In practice, such visual cues are encoded using a cost function that is defined at every pixel location, and that depends of the color values or other derived properties occurring in its vicinity. Similarly, the plausibility constraints can be approximately solved using a different type of visual cues that depends on the editing decisions made for other nearby pixels. This type of cues often encourages taking editing decisions that are compatible with its vicinity. If defined appropriately, they can emulate the type of consistency that humans expect in natural images. Once both types of visual cues are selected, they can be merged in a single cost function whose minima correspond to the desired editing result.

This strategy corresponds to a Bayesian inference problem with a Markov-random-field prior, which can be approximated efficiently using graph cuts (see Sec. 2.1). These approximations can be obtained in polynomial time, with warranties on the minimum distance to the global optima that can be achieved [Boykov01]. After its introduction to computer vision three decades ago [Greig89], this strategy has been successfully applied to a wide variety of problems in image and video editing [Kwatra03, Agarwala04, Rother05,

Eden06, Kolmogorov08, Pritch09, Bai09, Hu10, Schoenemann12]. In this thesis, I propose methods that advance the state-of-the art of video and image sequence editing by applying this strategy to two standing problems: Removing objects from video sequences, and removing ghosting artifacts from image sequences. On overview of these problems is given next.

1.1 Overview

This thesis proposes new solutions for two challenging problems in image and video editing: The first editing requires the removal of ghosting artifacts from high dynamic range (HDR) images that are reconstructed from a low dynamic range (LDR) image sequences (Part I); the second editing requires the removal of unwanted objects from video sequences taken with static and moving cameras (Part II). The proposed solutions enable high-quality editings that were not possible before. This is achieved without requiring a semantic understanding of the scene thanks to the inclusion of new key ingredients such as camera noise models and Bayesian inference frameworks.

1.1.1 Part I: Editing of Exposure Sequences

In the first part of this thesis, I propose methods for improving the editing of high dynamic range images that are recovered from *exposure sequences*, i.e., sequences where every image is taken with a different exposure time. Let me motivate this problem using a real-world application: HDR images are often used to simulate the lighting of real-world scenes, in a way that it enables the rendering of virtual objects under the scene's illumination so they can be merged with the real scene in a plausible way, or it makes it possible to create realistic virtual scenes by using complex, real-world lighting. For this purpose, it is necessary to acquire environment maps, i.e., 360 degree images of the scene. These environment maps can be recovered by combining several photographs at different exposure time. This is often required as the dynamic range of digital cameras is not sufficient to properly capture the light of many natural scenes. However, when photographing in uncontrolled public spaces, there might be moving scene elements that need to be removed before a proper environment map can be recovered. This situation is addressed on Chapter 3, where moving objects are detected and excluded from the reconstruction of high dynamic range images. This problem is known as *de-ghosting*.

For addressing this problem, I take advantage of a noise model for CCD/CMOS sensors in order to predict the magnitude of the noise in the input images, so that moving objects can be reliably detected. For de-ghosting, the proposed algorithm does not assume any semantic information of the scene, such as the extent of moving objects, or their correspondence

between images. For this reason, this method can be considered to be scene-independent, i.e., it does not make any assumptions about the actual content of the scene. The evaluation of the proposed method shows that it has superior de-ghosting performance when compared with related approaches in the literature. Additionally, Chapter 4 shows that the predicted noise also helps improving the performance of three other related tasks: The generation of exposure sequences that achieve a minimum signal-to-noise ratio (SNR), the de-noising of HDR images without affecting the image content, and the reconstruction of HDR images with optimal SNR. The latter method is demonstrated to produce HDR images with higher SNR than any other existing method.

1.1.2 Part II: Editing of Video Sequences

In the second part of this thesis, I make a transition from exposure sequences to video sequences, and I propose methods for performing advanced video editing tasks such as restoring damaged regions of videos and removing unwanted scene elements from them. As mentioned before, this editings are required in several scenarios. For instance, when a movie needs to be shot in a public place, it is often the case that unplanned objects like pedestrians or cars appear in the scene. Also, crew members that need to be in the shot need to be removed in post-processing. To perform this task, I take advantage of the observation that video sequences often contain a high amount of *redundant* information. This redundancy is exploit to restore the scene behind the unwanted scene elements by reusing other views available in different video frames. This principle is demonstrated in two closely related problems: The removal of scene elements that occlude other *dynamic elements* in the scene from videos taken with *static cameras* (Chapter 5), and the removal of scene elements that occlude other *static elements* in the scene from videos taken with *dynamic cameras* (Chapter 6). The proposed methods do not make any assumptions about the type of objects in the scene, and therefore, they can be considered to be scene-independent. These methods are experimentally demonstrated to produce higher quality editings than state-of-the-art methods, and additionally, they are shown to extend the range of camera motions that can be handled.

1.2 Claims

In summary, in this thesis I claim that it is possible to perform plausible high level editings on image and video sequences without modeling or making strong assumptions about the content of the scene. This is possible by defining local visual cues that approximate the editing requirements and plausibility constraints. We provide evidence using two different types of

editings: Removal of ghosting artifacts from exposure sequences, and removal of objects from video sequences. The proposed solutions take advantage of global optimization strategies to achieve results that look plausible to humans. It is our hope that these advances help the people working in the visual arts to focus their energy in other more creative tasks.

1.3 Contributions

The editing methods presented in this thesis have been presented in international research conferences and journals [Granados10, Granados12b, Granados12a, Granados13]. This work presents an extended revision of these methods. The key contributions are:

- A simple but robust image difference test for detecting differences between photographs of the same scene taken under large differences of exposure using a new camera noise calibration method. Based on this test, a new method is proposed for reconstructing plausible HDR images of dynamic scenes (Chapter 3). The resulting method has the best ghosting detection accuracy among existing competing methods and it is the first to work on highly clutter dynamic scenes.

- A simple method for reconstructing and denoising HDR images with optimal signal-to-noise ratio based on a camera noise model (Chapter 4). According to our experimental evaluation, and preliminary third-party evaluations [Aguerrebere12], the proposed reconstruction method obtains the best signal-to-noise ratio among the methods available in the literature.

- A new method for removing objects from video sequences that is able to complete the motion of other occluded dynamic objects by using redundant information in the video (Chapter 5). The proposed method produces more plausible results than state-of-the art methods, and it is the first to be shown to create production-quality inpaintings of dynamic objects on high resolution videos.

- A new algorithm for aligning images based on a piece-wise planar assumption about the geometry of the scene. Using this building block, a new method is proposed for removing objects from video sequences that occlude other static objects (Chapter 6). This method is able to cope with camera motion without needing to resort to complex and error-prone models of the camera position and scene geometry. It is shown to perform well even in scenarios where the camera motion is hard to estimate. The proposed method is the first to show results on videos with such camera motion.

- In general, this work presents further evidence that the semantic content of video and image sequences can be modified in a plausible way without having to construct models for the scene content. This is made possible by the application of well established frameworks for global energy minimization and Bayesian inference.

CHAPTER 2

Preliminaries

This chapter provides an account of the energy minimization methods and noise models that are the basis of the image and video editing methods proposed in this thesis.

In Sec. 2.1, a summary of the energy minimization methods based in graph cuts is provided. These minimization method is fundamental piece of the HDR de-ghosting method proposed in Chapter 3, and of the video inpainting methods proposed in Chapter 5 and Chapter 6.

In Sec. 2.2, a summary of the model used for predicting the noise in digital cameras is presented. This noise model is the basis of the HDR image editing methods proposed in Chapter 3 and Chapter 4.

2.1 Energy Minimization in Computer Vision

Many problems in early vision require the estimation of a spatially varying quantity, such as pixel intensity (image de-noising), pixel disparity (stereo), or pixel displacement (motion estimation). In many cases, such quantities can be assumed to be piece-wise smooth or piece-wise constant, where the discontinuities normally occur at the boundary of the objects in the scene. These estimated quantities should comply as much as possible with the observed data, while preserving the properties that correct results for each problem are expected to satisfy.

This type of early vision problems can be naturally expressed in terms of minimizing a global energy function. The definition of such an energy function enables the precise expression of the properties of the desired re-

sults. Formally, the objective is to obtain a function $F(p)$ that minimizes an energy functional of the form

$$\mathcal{E}(F) = \mathcal{E}_{\text{data}}(F) + \mathcal{E}_{\text{prior}}(F), \tag{2.1}$$

where $\mathcal{E}_{\text{data}}$ measures the deviation from the observed data $\mathcal{X}(p)$, and $\mathcal{E}_{\text{prior}}$ measures the level of disagreement with the prior assumptions about the properties that suitable solutions $F(p)$ should satisfy.

In general, the data term $\mathcal{E}_{\text{data}}$ follows the form

$$\mathcal{E}_{\text{data}}(F) = \int_{\mathcal{I}} D(F(p), \mathcal{X}(p))\mathrm{d}p, \tag{2.2}$$

where the function D measures how well the value $F(p)$ is supported by the observed data $\mathcal{X}(p)$ over the image domain $\mathcal{I}$.

For instance, for the problem of image de-noising, a natural choice for this function is the squared difference $D(F(p), \mathcal{X}(p)) = (F(p) - \mathcal{X}(p))^2$.

On the other hand, for the same problem of image de-noising, the prior term could follow the form $\mathcal{E}_{\text{prior}} = \int_{\mathcal{I}} \Psi(|\nabla F(p)|^2)\mathrm{d}p$, where Ψ is a monotonically increasing function that penalizes large gradient magnitudes in the de-noised image F. Depending on the *importance* or weight assigned to each of the two energy terms, one can sacrifice fidelity to the original image for smoothness in the de-noised result, and vice versa. Note that the choice of $\mathcal{E}_{\text{prior}}$ has an important impact on the type of minima that are obtained. For instance, if the prior term encourages solutions that are smooth everywhere, i.e. for $\Psi(s^2) = s^2$, the resulting functional is convex and a global minimum can be easily obtained. However, this choice leads to poor results at the boundary of objects, where the solution is generally not smooth. If the objective is to also preserve image boundaries, functions that selectively penalize gradients depending on their magnitude could be used. For instance, the function $\Psi(s^2) = \lambda\sqrt{1 + \frac{s^2}{\lambda^2}}$ avoids penalizing large image gradients, while still penalizing smaller gradients that are likely caused by noise. The class of priors that preserve object boundaries are called *discontinuity preserving* functions. However, this property generally comes at the price of non-convexity, which makes the computation of a global minimum infeasible.

There exist several minimization methods available in the literature that can be applied depending on the particular structure of the energy function. For instance, if F is continuous, variational methods can be applied. These methods use the Euler-Lagrange equations of the energy in order to characterize solutions located at local minima; this strategy was introduced to computer vision by Horn and Schunck [Horn81]. On the other hand, the minimization of discrete energy functions is a well-studied topic in the field of combinatorial optimization The next section describes the minimization methods for discrete energy functions applied in this thesis, and their relation to Bayesian inference.

2.1.1 Energy Minimization as Bayesian Inference

In the context of Bayesian inference, Bayes' rule can be applied to estimate the likelihood of a *model* F given observations $\mathcal{X}$. This likelihood is derived as

$$\Pr(F|\mathcal{X}) = \frac{\Pr(\mathcal{X}|F)\Pr(F)}{\Pr(\mathcal{X})}, \tag{2.3}$$

where $\Pr(\mathcal{X}|F)$ is the probability distribution of a sample $\mathcal{X}$ given the model F, and $\Pr(F)$ is the prior probability distribution of the model. Often, it is required to find the model $\hat{F}$ that best explain the observations. If the prior $\Pr(F)$ is available, this model can be estimated as the mode of the posterior probability distribution $\Pr(F|\mathcal{X})$, i.e., by obtaining the estimate

$$\begin{aligned} \hat{F}_{\mathbf{MAP}} &= \arg\max_F \Pr(F|\mathcal{X}) \\ &= \arg\max_F \Pr(\mathcal{X}|F)\Pr(F). \end{aligned} \tag{2.4}$$

The resulting mode is known as the maximum a posteriori probability (MAP) estimate of the distribution. Note that, when the prior $\Pr(F)$ is not available or it is assumed to be constant, this method is equivalent to a maximum likelihood estimation.

In computer vision, a common strategy is to represent both the observed and desired values at every pixel in an image or video as random variables, i.e., by defining $\mathcal{X} = \{\mathcal{X}_1, \ldots, \mathcal{X}_n\}$ as the set of observations $\mathcal{X}_p$ at each pixel p in the domain $\mathcal{I}$, and $F = \{F_1, \ldots, F_n\}$ as the desired value to be estimated at every pixel, which is is obtained by maximizing the posterior probability $\Pr(F|\mathcal{X})$. The likelihood $\Pr(\mathcal{X}|F)$ is defined according to the domain of the problem.

For illustration, in the problem of image de-noising, the likelihood is modeled as the probability of the observed pixel value, assuming that the true distribution of F is known. This likelihood is defined according to a noise model for the image formation process. This noise is usually modeled as additive zero-mean Gaussian noise where the variance is a hyper-parameter. As we will discuss in Chapter 4, this assumption is not adequate, but nevertheless it is very common in vision.

On the other hand, the probability distribution $\Pr(F)$ should encode any prior knowledge regarding the distribution of the desired results. In low-level vision problems, these priors are represented using the Markov property as constraints that depend on the context of the pixel. This property requires that the probability of a given variable F_p depends only on the realization of the incident variables. In images, the incidence relation is defined by the adjacency relation on the lattice defined over the pixels p in the domain $\mathcal{I}$. This relation is represented in a neighborhood system $\mathcal{N} = \{\mathcal{N}_1, \ldots, \mathcal{N}_n\}$ that contains the set of pixels $\mathcal{N}_p$ adjacent to every pixel p. Using this notation,

the Markov property can be expressed as $\Pr(F_p|F \setminus \{F_p\}) = \Pr(F_p|F_{\mathcal{N}_p})$. When this property is satisfied, the random variable F and the neighborhood system $\mathcal{N}$ constitute a Markov random field (MRF).

Before the MRF prior $\Pr(F)$ can defined, the concept of clique needs to be introduced: A set of pixel locations is called a *clique* if it corresponds to a maximally connected sub-graph according to the adjacency relationship defined by the neighborhood system $\mathcal{N}$. Using this definition, the probability of the resulting MRF is given by

$$\Pr(F) = \frac{1}{Z} \prod_{c \in C} \phi_c(F), \tag{2.5}$$

where ϕ_c is the potential function for each clique $c \in C$, and Z is a normalization constant. According to Hemmerley-Clifford theorem [Besag74], it is possible to completely specify $\Pr(F)$ by only defining the potential functions of the maximal cliques, provided that $\Pr(F) > 0$. The resulting probability has the form

$$\Pr(F) = \frac{1}{Z} \exp\left(- \sum_{c \in \mathcal{C}} V_c(F) \right), \tag{2.6}$$

where $\phi_c(F) = -\log(V_c(F))$, and $\mathcal{C}$ is the set of maximal cliques. This general framework was introduced to computer vision by Geman and Geman for the problem of image de-noising [Geman88].

For the case where the neighborhood system contains only pairs of pixels adjacent in $\mathcal{I}$, the set of maximal cliques $\mathcal{C}$ is equivalent to $\mathcal{N}$. In this case, the clique potentials have the form $V_{i,j}(F_p, F_q)$, where i, j are the pixel locations and F_p, F_q the assumed true values. Now, assuming that the elements of F are independent, the likelihood $\Pr(\mathcal{X}|F)$ can be approximated as $\Pr(\mathcal{X}|F) = \prod_p \Pr(\mathcal{X}_p|F_p)$. In addition, when this likelihood follows a Gaussian distribution, it can be expressed as $\Pr(\mathcal{X}_p|F_p) = K \exp(-D_p(F_p))$, where K is a constant. Following these assumptions, and taking an MRF prior, the MAP estimate from Eq. 2.4 can be derived as

$$\begin{aligned} \hat{F}_{\mathbf{MAP}} &= \arg\max_F \exp\Big(- \sum_{i \in \mathcal{I}} D_p(F_p) \Big) \exp\Big(- \sum_{(p,q) \in \mathcal{N}} V_{p,q}(F_p, F_q) \Big), \\ &= \arg\min_F \sum_{i \in \mathcal{I}} D_p(F_p) + \sum_{(p,q) \in \mathcal{N}} V_{p,q}(F_p, F_q). \end{aligned} \tag{2.7}$$

For clarity, note that the functions $D_p, V_{p,q}$ have as implicit arguments the observations $\mathcal{X}_p$, and $\{\mathcal{X}_p, \mathcal{X}_q\}$, respectively.

In summary, energy minimization for vision problems can be cast as Bayesian inference with a Markov random field prior, where inference is approximated by MAP estimates. After its introduction to computer vision, this approach has been applied extensively in the field. In particular, there are very efficient methods for MAP inference based on graph

cuts [Boykov01]. These methods are applicable when the measurable set of F is discrete, provided that the clique potentials satisfy certain properties. These methods and the required conditions for efficient inference are the described in the following sections.

2.1.2 Minimization of Discrete Functionals

In the discrete setting, the possible values of F_p are defined by a finite set of labels $\mathcal{L}$. In this setting, the function $F : \mathcal{I} \to \mathcal{L}$ is called a *labeling*. Therefore, the optimization task can be seen as estimating a value $F_p \in \mathcal{L}$ for every pixel $p \in \mathcal{I}$, such that the corresponding energy $\mathcal{E}(F)$ is minimized. This energy can have the form

$$\mathcal{E}(F) = \sum_{p \in \mathcal{I}} D_p(F_p) + \sum_{(p,q) \in \mathcal{N}} V_{p,q}(F_p, F_q), \tag{2.8}$$

where $\mathcal{E}_{\text{data}}$ is defined by the unaware function D_p that measures the disagreement with the observed data $\mathcal{X}$, and the $\mathcal{E}_{\text{prior}}$ is defined by the binary function $V_{p,q}$ that measures the suitability of assigning labels F_p, F_q to adjacent pixels p, q.

Naturally, the definition of $V_{p,q}$ determines the type of prior assumed on the labelings. Instances of commonly used priors include

$$\text{the truncated quadratic difference } V_{p,q}(\alpha, \beta) = \max(k, |\alpha - \beta|^2), \tag{2.9}$$

$$\text{the truncated absolute difference } V_{p,q}(\alpha, \beta) = \max(k, |\alpha - \beta|), \text{ and} \tag{2.10}$$

$$\text{the } \textit{Potts} \text{ model } V_{p,q}(\alpha, \beta) = k \cdot \mathbb{1}_{\{\alpha \neq \beta\}}, \tag{2.11}$$

where α, β are labels in $\mathcal{L}$, and $\mathbb{1}_{\{\cdot\}}$ denotes the indicator function. The constant k is a scalar that bounds the maximum possible energy contributed by the function. The truncated difference functions imply a piecewise smooth prior, i.e., labelings that have locally smooth clusters, whereas the Potts model implies a piecewise constant prior, i.e., labelings containing clusters of constant value.

Energy minimization is known to be NP-hard [Veksler99], even for the simplest potential, i.e., the Potts model (Eq. 2.11). For this reason, only approximated solutions to the minimization problem can be expected. A common approach is to seek for local minima in a forward stage-wise greedy fashion. However, the main drawback of this strategy is that it can converge to local minima that are arbitrarily far from the global optima. This makes it very difficult to decide whether a wrong solution corresponds to labeling that is far from the optima, or to an energy function that does not correctly represent the constraints of the problem at hand.

For overcoming this difficulty, Boykov et al. [Boykov01] consider the optimality properties of a given approximation. A local minimum is a function

F such that $E(F) < E(F')$ for every F' *close to* F. The confidence on such a solution increases with the order of possible functions considered in its neighborhood. For instance, the simulated annealing method provides local minima where the energy cannot be decreased by a *standard move*, i.e. by changing a the label of a single pixel at a time. The confidence on such a result is low since the number of labelings within a standard move is linear in the number of pixels. The simulated annealing method was introduced to computer vision by Geman and Geman [Geman88]. However, significantly larger moves are considered by α-*expansions*, which are described in the next section.

2.1.3 Minimization of Multi-label Functionals

Boykov et al. [Boykov01] developed two energy minimization algorithms based in graph-cuts that produce a different type of local minima (graph-cuts are described in Sec. 2.1.4). These algorithms obtain labelings that are minima with respect to two types of large moves: α-β-*swaps* and α-*expansions*. In contrast to standard moves, these moves cover an exponentially large set of labelings, since they allow more than one pixel to change label at each step. The first type, the α-β-swap, allows any p with label $F_p = \alpha$ to move to label $F_p = \beta$, and vice versa. The second type, the α-expansion, allows any pixel to be assigned the label $F_p = \alpha$. It can be shown that standard moves are a special case of α-β-swaps and α-expansions.

The minimization algorithm for α-β-swaps and α-expansions is structurally equivalent, and the latter is called the *expansion move* (see Algorithm 1). It proceeds as follows: First, the output labeling is initialized. Then, for every pair of labels (α, β), or for each label α, it minimizes the energy with respect to the current α-β-swap, or α-expansion, respectively; this operation is called a *cycle*. Within each cycle, it proceeds as follows: First, it computes the labeling with minimum energy with respect the current move; this is the main step of the algorithm. If the energy is successfully decreased, the labeling replaces the current solution. The algorithm terminates after the first cycle that does not decrease the energy. In general, the resulting labeling does not change significantly with respect to the initialization, due to the use of larger moves.

Unlike with α-β-swaps, the expansion move algorithm with α-expansions provides an optimality guaranty in terms of the distance to the global minimum. This guaranty states that for every approximate solution F, the inequality

$$E(F) \leq 2cE(F^*) \tag{2.12}$$

holds, where F^* is a global minimum, and c is the constant

$$c = \max_{p,q \in \mathcal{N}} \left(\frac{\max_{\alpha \neq \beta \in \mathcal{L}} V_{p,q}(\alpha, \beta)}{\min_{\alpha \neq \beta \in \mathcal{L}} V_{p,q}(\alpha, \beta)} \right) \tag{2.13}$$

```
F=arbitrary initial labeling;
repeat
    success=false;
    foreach label α ∈ 𝓛 do
        f̂=argmin E(f̂′) among all F′ within one α-expansion of F;
        if E(f̂) < E(F) then
            F=f̂;
            success=true;
        end
    end
until !success;
```

Algorithm 1: Expansion move with α-expansions

that depends on the prior potentials only. For instance, for the Potts model (Eq. 2.11), this constant is given by $c = 1$; it follows that the expansion move algorithm will compute labelings that have at most twice as much energy as the global minimum.

The key step in the algorithm, i.e., computing the α-expansion, corresponds to a binary label optimization problem. This can be performed in polynomial time using the graph-cut/min-flow algorithm, which is described in the next section.

2.1.4 Minimization of Binary Functionals

The core of energy minimization using the expansion move algorithm is the α-expansion step. This step can be cast as a binary labeling problem where each pixel either keeps its current label $F_p = \gamma$ or moves to the label $F_p = \alpha$, in such a way that the energy is decreased. This problem can be solved by computing a minimum cut on a graph representing the energy $\mathcal{E}$. This was first proposed by Grieg et al. [Greig89] in the context of computer vision. The representation of a labeling energy using a graph is discussed next, and the algorithm for graph construction, and the definition of minimum cut are provided afterward.

Graph Representability

Kolmogorov and Ramin [Kolmogorov04] study the set of energy functions over binary labelings that can be minimized via graph cuts. An energy function of n binary variables is called *graph representable* if there exists a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ with terminals $s, t \in \mathcal{V}$ and a subset of vertices $\{v_1, \ldots, v_n\} \subset \mathcal{V} \setminus \{s, t\}$ such that, for any configuration of the binary variables $\mathcal{B} = \{\mathcal{B}_1, \ldots, \mathcal{B}_n\}$, the energy $E(\mathcal{B})$ is equal to the cost of the *minimum s-t-cut* among all cuts $\mathcal{C} = \{\mathcal{S}, \mathcal{T}\}$ of $\mathcal{G}$. In this representation, a node $v_i \in \mathcal{S}$

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} E^{i,j}(0,0) & E^{i,j}(0,1) \\ E^{i,j}(1,0) & E^{i,j}(1,1) \end{pmatrix} \equiv \begin{pmatrix} V_{p,q}(F_p, F_q) & V_{p,q}(F_p, \alpha) \\ V_{p,q}(\alpha, F_q) & V_{p,q}(\alpha, \alpha) \end{pmatrix}$$

Table 2.1: The expansion move algorithm transforms a multiple label assignment problem into a binary assignment one. At each iteration, the current labels F_p, F_q are encoded by zero, and the expanding label α is encoded by one.

if $\mathcal{B}_i = 0$, or $v_i \in \mathcal{T}$ if $\mathcal{B}_i = 1$. The exact definition of minimum cut is provided below in the section regarding the Graph-Cut/Max-flow problem.

In particular, they approach the graph representability of the class $\mathcal{F}^2$ of functionals of the form

$$E(\mathcal{B}) = \sum_i E^i(\mathcal{B}_i) + \sum_{i<j} E^{i,j}(\mathcal{B}_i, \mathcal{B}_j), \tag{2.14}$$

which corresponds to class of functionals defined as the sum of functions of up to two binary variables $\mathcal{B}_i \in \{0, 1\}$. Within this class, the functions $E^{i,j}$ satisfying the inequality

$$E^{i,j}(0,0) + E^{i,j}(1,1) \leq E^{i,j}(0,1) + E^{i,j}(1,0) \tag{2.15}$$

are called *regular* or *submodular*. Their main contribution states that an energy function $E \in \mathcal{F}^2$ is *graph representable* if and only if each binary term $E^{i,j}$ is regular. Note that there is no restriction on the sign of the energy function of the individual terms.

In the expansion move algorithm, every pixel either keeps its current label or changes it to α on each move. This can be encoded using binary labels, e.g. zero represents the current label, and one represents α. Therefore, for energies of the form defined in Eq. 2.8, each iteration of the expansion move can be performed by minimizing an energy function in the class $\mathcal{F}^2$. The proper encoding is illustrated in Fig. 2.1 for functions $E^{i,j}$. The prior potentials $V_{p,q}$ need to be chosen such that the corresponding binary term $E^{i,j}$ is regular. Given two labels β, γ and the expanding label α, this condition is satisfied if the inequality

$$V_{p,q}(\beta, \gamma) \leq V_{p,q}(\beta, \alpha) + V_{p,q}(\alpha, \gamma) \tag{2.16}$$

holds for every pair of neighbors $(p, q) \in \mathcal{N}$. Note that the cost $V_{p,q}(\alpha, \alpha)$ is assumed to be zero since no discontinuity is introduced in the labeling. In particular, this triangular inequality is satisfied when $V_{p,q}$ is a metric in the label set.

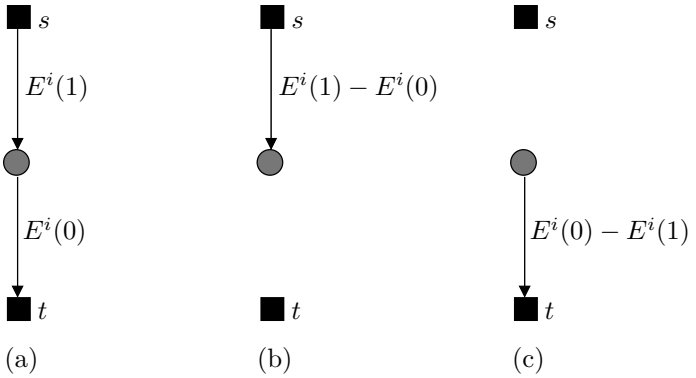


Figure 2.1: The energy of unary terms E^i represented through a graph. (a) Representation for $E^i \geq 0$. (b) Representation for $E^i(1) > E^i(0)$. (c) Representation for $E^i(0) > E^i(1)$.

Graph Construction

Kolmogorov and Ramin [Kolmogorov04] also provide an algorithm for constructing graphs whose minimum cut minimizes a binary energy function that is graph representable. The graph $\mathcal{G}$ will have a set of nodes $\mathcal{V} = \{s, t, v_1, v_2, \ldots, v_n\}$, where n is the number of binary variables. The source s corresponds to label zero ($\mathcal{B}_i = 0$), and the sink t to label one ($\mathcal{B}_i = 1$). The set of edges $\mathcal{E}$ is defined according to the functions in E. An edge connecting a non-terminal node v_i and a terminal node is called *t–link*; an edge connecting two non-terminal nodes is called an *n–link*.

First, consider unary terms E^i that depend on a single variable $\mathcal{B}_i$. The objective is to define the edges in $\mathcal{G}$ such that the energy minimum of E^i corresponds to a minimum cut of $\mathcal{G}$. A straightforward solution corresponds to adding to the graph the edge (s, v_i) with weight $E^i(1)$, and the edge (v_i, t) with weight $E^i(0)$. In this way, if (s, v_i) is in the minimum cut, then v_i belongs to the sink partition, and $\mathcal{B}_i$ is assigned the label one. The case of (v_i, t) is analogous. However, this restricts the terms E^i to be always positive (this is a constraint of the minimum cut algorithm described below). To lift this restriction, one can exploit the fact that energy minima are unchanged by the addition of a constant, and hence, one can subtract the value $\min\{E^i(0), E^i(1)\}$ from the weight of both edges. This results in a least one edge with zero weight, which is removed from the graph, and a second edge with non-negative weight. Therefore, it is only required to add a single edge per unary term, i.e., (s, v_i) with weight $E^i(1) - E^i(0)$ if $E^i(0) < E^i(1)$, or (v_i, t) with weight $E^i(0) - E^i(1)$ otherwise. This is illustrated in Fig. 2.1.

Now let us consider the binary terms $E^{i,j}$ that depend on two binary variables $\mathcal{B}_i, \mathcal{B}_j$. For convenience, such terms can be reformulated in one of the forms presented in Fig. 2.2. Since the terms $E^{i,j}$ are regular, the

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} A-C & A-C \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} C-D & 0 \\ C-D & 0 \end{pmatrix} + \begin{pmatrix} 0 & B+C-A-D \\ 0 & 0 \end{pmatrix} + D$$

(a) For $A > C$, $C > D$

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} A-C & A-C \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & D-C \\ 0 & D-C \end{pmatrix} + \begin{pmatrix} 0 & B+C-A-D \\ 0 & 0 \end{pmatrix} + C$$

(b) For $A > C$, $D > C$

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ C-A & C-A \end{pmatrix} + \begin{pmatrix} C-D & 0 \\ C-D & 0 \end{pmatrix} + \begin{pmatrix} 0 & B+C-A-D \\ 0 & 0 \end{pmatrix} - C + A + D$$

(c) For $C > A$, $C > D$

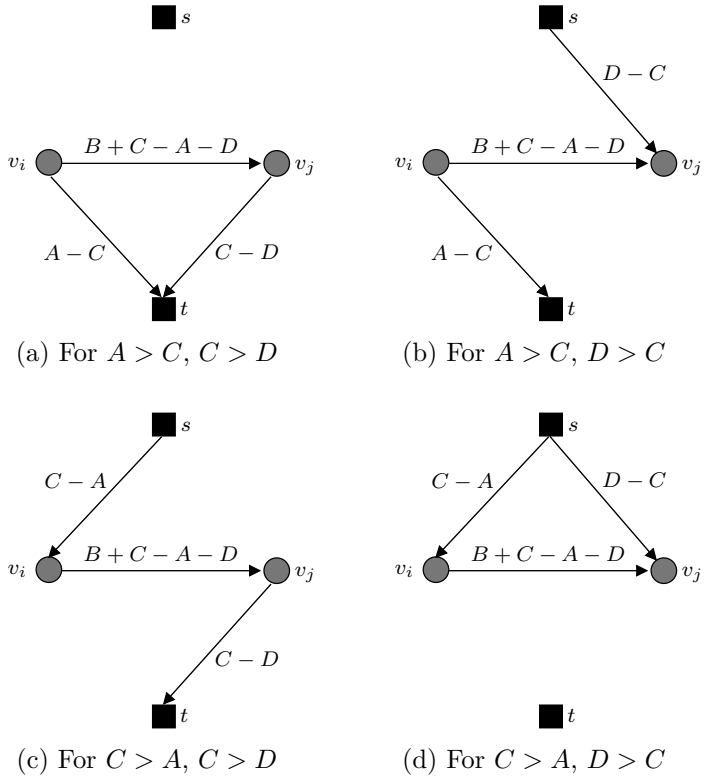
$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ C-A & C-A \end{pmatrix} + \begin{pmatrix} 0 & D-C \\ 0 & D-C \end{pmatrix} + \begin{pmatrix} 0 & B+C-A-D \\ 0 & 0 \end{pmatrix} + A$$

(d) For $C > A$, $D > C$k

Figure 2.2: Reformulation of the terms $E^{i,j}$ for graph representation.

expression $E^{i,j}(0,1)+E^{i,j}(1,0)-E^{i,j}(0,0)-E^{i,j}(1,1)$ is always non-negative (see Eq. 2.15). In Fig. 2.2, each matrix form is decomposed into three matrices plus a scalar. The scalars do not induce edges to the graph. The cost of each possible label assignment to v_i, v_j can be represented using three edges. For instance, the form Fig. 2.3a follows the correspondences shown in Fig. 2.2a. For this form, $A > C$ and $C > D$. For instance, if we set $\mathcal{B}_i = 0$ and $\mathcal{B}_j = 0$, it implies that a minimum cut passes through the edges (v_i, t), (v_j, t) of the graph. The resulting cut has cost $(A-C)+(C-D) = A-D$. Adding the scalar D results into the original cost $E^{i,j}(0,0)$. As another example, assigning $\mathcal{B}_i = 0$ and $\mathcal{B}_j = 1$ implies cutting the edges (v_i, t) and (v_j, s). This cut has cost $(A-C)+(B+C-A-D) = B-D$, which corresponds to the desired cost $E^{i,j}(0,1)$ plus the scalar D.

The additivity theorem [Kolmogorov04] states that the sum of two graph representable functions is itself graph representable. This allows us to construct a single graph to represent the energy of the complete binary labeling E. The final graph is obtained by adding up the edge weights computed for each of the terms $E^i, i \in \mathcal{I}$ and $E^{i,j}, (i,j) \in \mathcal{N}$. A minimum-cut of this graph corresponds to the label assignment that leads to a labeling with minimum energy. The minimum-cut algorithm is described next.



(a) For $A > C$, $C > D$ (b) For $A > C$, $D > C$

(c) For $C > A$, $C > D$ (d) For $C > A$, $D > C$

Figure 2.3: Graph representation for the energy of binary terms $E^{i,j}$.

Graph-Cut/Max-Flow Problem

Now, the formal definition of the minimum cut problem is provided. Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a directed weighted graph, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ is the set of edges. Let $s, t \in \mathcal{V}$ be two special nodes called the *source* and the *sink* respectively. These special nodes are referred as *terminals*. Each directed edge $(p, q) \in \mathcal{E}$ is assigned a non-negative weight $w(p, q)$. An *s–t–cut* $\mathcal{C} = \{\mathcal{S}, \mathcal{T}\}$ in $\mathcal{G}$, or *cut* for short, is a partitioning of $\mathcal{V}$ into two disjoint sets $\mathcal{S}$ and $\mathcal{T}$ such that $v \in \mathcal{V}$ and $t \in \mathcal{T}$. The cost of a cut $\mathcal{C}$, denoted by $[\mathcal{C}]$, is defined as the sum of the weights $w(p, q)$ of the *boundary edges* (p, q) which satisfy $p \in \mathcal{S}$ and $q \in \mathcal{T}$. Note that reverse edge weights $w(q, p)$ are not accounted in the cost. The *minimum cut* problem is defined as computing a cut with minimum cost.

The minimum cut problem can be solved efficiently. A fundamental theorem by Ford and Fulkerson [Ford62] states that it is equivalent to compute the *maximum flow* between the terminals. This is a well-studied combinatorial optimization problem, and there exists several algorithms for obtaining a solution in polynomial time. See [Boykov04] for a comparison of max-flow algorithms applied to energy minimization in computer vision.

The relation between max-flow and min-cut can be understood intuitively. The maximum flow from the source to the sink is bounded by the capacity of those bottle neck edges that would become saturated in a flow simulation. These edges become saturated precisely due to their limited capacity. This rationale illustrates why a minimum cut should pass through such edges with low capacity.

Handling of Non-regular Terms

In the previous sections, the properties that an energy function needs to satisfy in order to be effectively minimized using graph cuts were summarized. For the multi-label problem, the requirements boils down to the condition that binary terms are submodular, i.e., that they define a metric on the labeling set (see Eq. 2.16). However, some applications require to include non-metric prior potentials [Kwatra03, Agarwala04, Rother05]. In such situations, the expansion move algorithm can be still applied provided that the number of non-regular terms in the energy function is relatively small.

Rother et al. [Rother05] formalize the conditions under which the expansion move algorithm can handle non-regular terms. They classify such terms into *hard constraints*, and *soft constraints*. Hard constraints are non-metric terms of the form $H_{p,q} \in \{0, \infty\}$. They can be included in order to forbid undesired configurations in the result. In their work, they prove that the optimality guaranties of the expansion move algorithm (Eq. 2.12) still hold after including hard constraints, provided that the initial energy has finite value. No modifications to the algorithm are required. In the

other hand, soft constraints are non-metric terms of the form $V_{p,q} \in \mathbb{R}$. They invalidate the regularity condition (Eq. 2.15), and therefore introduce negative edge weights in the graph, which prevents the application of the maximum flow algorithm. In order to handle this situation, Rother et al. propose to *truncate* non-regular terms, i.e., to replace them with regular terms, and minimize the resulting energy function. Provided that unary terms remain unchanged, they demonstrate that if the truncation process follows certain conditions, expansions do not increase the labeling energy. Formally, let $\hat{V_{p,q}}$ be the truncated term, F be the initial labeling with $F_p = \beta$, $F_q = \gamma$, and α be the current expanding label. If for every $(p,q) \in \mathcal{N}$ it holds that $\hat{V_{p,q}}(\beta,\gamma) \leq V_{p,q}(\beta,\gamma)$, and $\hat{V_{p,q}}(F_p,F_q) \geq V_{p,q}(F_p,F_q)$ with $(F_p,F_q) \neq (\alpha,\alpha)$, then if F^* minimizes the modified energy function $\hat{E}$ then $E(F^*) \leq E(F)$. This theorem implies the following truncation procedure. If a term $V_{p,q}$ is not regular, i.e., $V_{p,q}(\beta,\alpha)+V_{p,q}(\alpha,\gamma) < V_{p,q}(\beta,\gamma)$, then one of the following operations is performed: To increase the energies $V_{p,q}(\beta,\alpha)$ and $V_{p,q}(\alpha,\gamma)$, or to decrease the energy $V_{p,q}(\beta,\gamma)$, until the inequality holds. In practice, the algorithm is likely to provide suitable local minima in situations where most terms are regular.

2.2 Noise Model for Digital Cameras

In Chapter 3 and Chapter 4, I will present methods for editing and processing high dynamic range images that are reconstructed from exposure sequences. As it will be shown on these chapters, the editing process can take advantage of the possibility to predict the magnitude of the noise of the color values observed in a digital photograph. Several methods exist for characterizing noise in digital cameras based on CCD/CMOS sensor technologies [Healey94, Janesick01, Reibel03]. In the first part of this thesis, the camera noise model presented in [Janesick01] is applied, as it accounts for the noise sources relevant to the problem of high dynamic range image processing.

The noise induced on photographs by digital imaging sensors is a combination of several sources. These noise sources are well studied in the field of optics and photonics. In general, noise sources can be classified into two categories: Temporal sources, and spatially-varying sources. The most relevant sources of each type are described next.

2.2.1 Sources of Temporal Noise

The first category, *temporal noise*, contains sources that cause the color of a pixel to change between photographs whose illumination and acquisition parameters are otherwise identical. In this work, the following sources of

temporal noise are considered: Photon shot noise, dark-current shot noise, and readout noise.

Photon Shot Noise (PSN)

Photon shot noise corresponds to the uncertainty that is intrinsic to the process of light emission. The number of photons emitted by an object (and arriving to the camera sensor) in a given time slack is well modeled by a Poisson distribution, where the expected value is equal to the variance. This number is known as the *exposure* $E = X\mathrm{t}$, which corresponds to the irradiance X integrated over the exposure time t. Since the exposure follows a Poisson distribution, we have $\mathrm{E}[E] = \mathbb{V}\mathrm{ar}[E]$. The uncertainty in the exposure is called *photon shot noise*.

At first, this could imply that photographs with long exposure, or that contain bright objects would suffer more from shot noise. However, HDR images normally represent the irradiance X at every pixel, which can be recovered by dividing the exposure by the exposure time, i.e., $X = \frac{E}{\mathrm{t}}$. It follows that the uncertainty of the irradiance estimate can be approximated as $\mathbb{V}\mathrm{ar}[X] = \frac{\mathbb{V}\mathrm{ar}[E]}{\mathrm{t}^2} \approx \frac{E}{\mathrm{t}^2} = \frac{X}{\mathrm{t}}$. For this reason, long exposures provide more reliable estimates of the irradiance in the scene as its uncertainty decreases with the exposure time.

Dark-current Shot Noise (DCSN)

In an ideal sensor, every photon that arrives to its photo-sensitive area *frees* a constant amount of electrons (just one for the visible spectrum). However, thermal energy causes some electrons to be freed without any incident photons involved. The effect of this additional charge is called *dark current*. The dark current is independent of the light intensity, and its magnitude depends on the temperature of the sensor, and on the exposure time. This noise source can be eliminated by reducing the temperature of the sensor. Analogous to PSN, the process of generation of thermo-electrons follows a Poisson distribution, whose uncertainty is called *dark-current shot noise*.

Readout Noise

In a sensor, the process of converting accumulated charge into digital values is disturbed by several other noise sources. These sources include *reset noise*, which occurs during charge-to-voltage transfer; *white noise* and *flicker noise*, which affects voltage amplification; and *quantization noise*, which occurs during analog-to-digital conversion. Please refer to [Janesick01] for an in-depth analysis of each of these sources. Since readout noise is a combination of several independent noise sources, it can be described by a Gaussian distribution. The readout noise is also independent of the light intensity.

2.2.2 Sources of Spatial Noise

The second category of noise, *spatial noise*, corresponds to sources that cause color differences between different pixels (located at different positions) that are exposed to identical light intensities. In the noise model used in this thesis, the following spatial sources are included: Photo-response non-uniformity, and dark-current non-uniformity.

Photo-response non-uniformity (PRNU)

In digital imaging sensors, there exist differences between the photosensitive area of different pixels. This occurs in spite of the high quality of the manufacturing process of CCD/CMOS sensors. These differences cause two given pixels to produce consistently different readings after being exposed to the same light intensity. The resulting differences in sensitivity are called *photo-response non-uniformity*. These non-uniformities can be modeled as a per-pixel gain factor. Therefore, the discrepancies caused by PRNU increase with the light intensity, and thus, this type of noise is more evident in brighter color values.

Dark-current non-uniformity (DCNU)

The amount of dark current varies between pixels due to temperature differences between pixels in the sensor. This variation is known as *fixed pattern noise (FPN)* or *dark current non-uniformity (DCNU)*. These non-uniformities can be modeled as a per-pixel bias. DCNU can be corrected by subtracting from each photograph a *dark frame*, i.e., an image acquired with no incident light (e.g. the lens covered) but otherwise identical camera settings, including the integration time and sensor temperature. For this reason, dark frames are best acquired alongside with every photograph. Most consumer cameras already contain provisions for subtracting dark frames from the acquired photographs, especially in long exposures.

2.2.3 Image Acquisition Model

We follow the image acquisition model described in [Janesick01]. This model can be used to estimate the irradiance falling onto the imaging sensor as a function of the digital output value produced by the camera.

Let t_i be the i-th exposure time of a photograph in an image sequence. Let $X_i(p)$ be the number of photo-induced electrons collected by the capacitor at pixel p per unit time on the image i; this is the value that can be estimated through the model, and it corresponds to a factor of the real irradiance value. Absolute irradiance values can be derived if the pixel area, and the quantum efficiency at for the particular wavelength are both known. Let $D_i(p)$ be the number of photon-electrons induced by dark current. And,

let $\mathrm{a}(p)$ be the pixel gain factor caused by the PRNU. During the exposure time, the pixel capacitor will collect

$$E_i(p) = \mathrm{t}_i \left(\mathrm{a}(p) X_i(p) + D_i(p)\right) \tag{2.17}$$

electrons; this value corresponds to the exposure. Assuming the exposure is known, the model predicts that the digital output value $V_i(p)$ at pixel p is generated as

$$V_i(p) = \mathrm{f}(E_i(p)), \tag{2.18}$$

where f is the *camera response* function that maps exposure values to digital values.

In this thesis, I assume that the camera response function is a linear function. This is a reasonable assumption, since first, the response of CCD/ CMOS sensors is close to linear, and second, many cameras allow to access the *raw* digital output, i.e., the output of the sensor before any potentially non-linear operations like demosaicing, white balancing, tone mapping, denoising, sharpening, or compression takes place in the camera. If the camera response is linear, the model for digital output values is given by

$$V_i(p) = \left[\mathrm{g} \cdot E_i(p) + N_R\right], \tag{2.19}$$

where g is called the *camera gain*, and N_R is a random variable that represents the readout noise, whose variance is denoted as σ_R^2, and whose mean is located at the black level value L_0, which corresponds to the offset of the analog-to-digital conversion. Lastly, the operator $[\cdot]$ represents the round-off operator that corresponds to the quantization occurring in the last stage of the analog-to-digital conversion.

Assuming the camera gain g, the PRNU a, and the exposure time t_i to be known, the variance prediction for $V_i(p)$ is given by

$$\sigma^2_{V_i(p)} = \mathrm{g}^2 \sigma^2_{E_i(p)} + \sigma_R^2, \tag{2.20}$$

where $\sigma^2_{E_i} = \mathrm{E}[E_i]$ accounts for the shot noise (both PSN and DCSN, well modeled by a Poisson distribution) and the non-uniformities (PRNU and DCNU), and σ_R accounts for the readout noise, including the quantization error.

In an analogous way, the predicted digital output in a dark frame $B_i(p)$ at pixel p, and its variance $\sigma^2_{B_i(p)}$ are given by

$$B_i(p) = \left[\mathrm{g} \cdot \mathrm{t}_i D_i(p) + N_R\right], \text{ and} \tag{2.21}$$

$$\sigma^2_{B_i} = \mathrm{g}^2 \sigma^2_{D_i} + \sigma_R^2, \tag{2.22}$$

respectively, where $\sigma^2_{D_i} = \mathrm{E}[\mathrm{t}_i D_i]$ accounts for the DCSN and DCNU, and σ_R^2 represents the readout noise and quantization.

Given observed pixels values $V_i(p)$ and $B_i(p)$, the irradiance X_i, and its variance σ_{X_i} can be derived from Eq. 2.19 and 2.21, and obtain the estimates

$$X_i(p) \approx \frac{V_i(p) - B_i(p)}{\mathrm{t}_i \cdot \mathrm{g} \cdot \mathrm{a}(p)}, \text{ with} \tag{2.23}$$

$$\sigma^2_{X_i(p)} \approx \frac{\sigma^2_{V_i(p)} + \sigma^2_{B_i(p)}}{\mathrm{t}_i^2 \mathrm{g}^2 \mathrm{a}(p)^2}. \tag{2.24}$$

Here, the quantization operator prevents obtaining an exact estimation.

Analogously, the dark current D_i and its variance can be predicted as

$$D_i(p) \approx \frac{B_i(p) - \mathrm{L}_0}{\mathrm{t}_i \cdot \mathrm{g}}, \text{ with} \tag{2.25}$$

$$\sigma^2_{D_i(p)} \approx \frac{\sigma^2_{B_i(p)} + \sigma^2_R}{\mathrm{t}_i^2 \mathrm{g}^2}. \tag{2.26}$$

2.2.4 Estimation of Noise Parameters

In order to estimate the irradiance $X_i(p)$, estimates for the following parameters are required: The black level L_0, the readout noise variance σ^2_R, the photo-response non-uniformity $\mathrm{a}(p)$, and the camera gain factor g. In addition, the saturation limit L_{sat} needs to be estimated, which specifies the maximum digital output of the camera. The estimation method for each parameter is described below, based on the strategies introduced in [Janesick01].

Black Level and Readout Noise

The black level and the readout noise variance can be estimated from a *bias frame*, i.e., an image acquired with zero integration time. In such a frame, virtually no photo- or thermo-electrons are collected, and hence, the output is perturbed only by signal independent noise. Assuming that the readout noise magnitude is decoupled from the pixel location, each pixel value can be considered as a sample of random variable of a distribution with mean L_0 and variance σ^2_R. Therefore, the parameters can be estimated as its expected mean and variance, i.e.

$$\mathrm{L}_0 = \mathrm{E}[B^b(p)] \approx \underset{p \in \mathcal{I}}{\mathbb{A}\mathrm{vg}}[B^b(p)], \text{ and} \tag{2.27}$$

$$\sigma^2_R = \mathrm{E}[(B^b(p) - \mathrm{L}_0)^2] \approx \underset{p \in \mathcal{I}}{\mathbb{V}\mathrm{ar}}[B^b(p)], \tag{2.28}$$

where $B^b(p)$ is the bias frame, and $\mathcal{I}$ is the set of pixel locations in the image. Note that the black level needs to be rounded off to an integer to fit the representation of the digital output values of the camera.

Saturation limit

Similarly to the black level, the saturation limit can be estimated as the rounded spatial mean of a *saturation frame*, i.e., an image where the sensor is exposed long enough so that every pixel reaches full-well capacity. This capacity is defined as the maximum number of electrons that the capacitor at every pixel can store. Then, the saturation limit L_{sat} can be estimated as

$$\mathrm{L}_{sat} = \mathrm{E}[V^s(p)] \approx \underset{p\in\mathcal{I}}{\mathbb{A}\mathrm{vg}}[V^s(p)] \tag{2.29}$$

where $V^s(p)$ denotes the saturation frame.

In general, it is reasonable to expect that all pixel values in the saturation frame be set at the maximum digital output value of the camera. However, this can be prevented by readout noise, if the maximum digital output plus the expected readout error is close to the digital value corresponding to full-well capacity, e.g. if $\mathrm{L}_{sat} + 6\sigma_R > \mathrm{f}(E^{fw})$, where E^{fw} denotes the full-well capacity. Therefore, to prevent potential failures in detecting saturated pixels, the saturation limit is set to $\mathrm{L}_{sat} - 6\sigma_{\mathrm{L}_{sat}}$, where $\sigma_{\mathrm{L}_{sat}}$ corresponds to standard deviation of the saturation frame.

Photo-response non-uniformity (PRNU)

An estimate of the PRNU can be obtained from a *flat field*, i.e., an photograph of a spatially uniform, narrow band light source, e.g. acquired using a diffuser and a bandpass wavelength filter. The PRNU factors are expected to follow a normal distribution with unit mean, and a small standard deviation (around 1% in practice). Therefore, the PRNU can be derived by dividing each pixel value $ff(p)$ in a flat field by the spatial frame average. However, in order to account for the effect of DCNU and readout noise, the corresponding dark frame needs to be subtracted, and then average several flat fields. This leads to the PRNU estimate

$$\mathrm{a}(p) = \frac{\mathrm{E}[ff(p) - B(p)]}{\mathbb{A}\mathrm{vg}_{p\in\Omega}[\mathrm{E}[ff(p) - B(p)]}, \tag{2.30}$$

where $B(p)$ denotes the dark frame corresponding to the flat field. In general, the exposure should be set such that the resulting output values are close to saturation, but not saturated, in order to reduce the effect of readout noise on the flat field.

Note that flat fields might dependent on the camera optics configuration. If lenses are present at the moment of calibration, the PRNU will also account for vignetting effects, and other lens distortions. If this is the desired effect, a separate flat field is necessary for each lens and focal length setting.

Camera gain

The camera gain is the factor that represents the proportion between the amount of charge stored at a pixel capacitor (i.e., the exposure E), and the final digital value output by the camera. Due to the quantum efficiency of CCD/CMOS sensors, this factor is wavelength dependent. In practice, this dependency can be ignored for the visible spectrum. For estimating the gain, the method described in [Janesick01] is followed. From Eq. 2.19, the following gain estimate can be derived

$$\mathrm{g} \approx \frac{\mathrm{E}[V_i(p)] - \mathrm{L}_0}{\mathrm{E}[E_i(p)]}. \tag{2.31}$$

If $V_i(p)$ is a flat field, each pixel can be considered a sample of the same random variable, so that the expectation can be approximated using spatial average, i.e. $\mathrm{E}[V_i(p)] \approx \mathbb{A}\mathrm{vg}_{p\in\mathcal{I}}[\mathit{ff}(p)]$. Recall that the exposure follows a Poisson distribution, so $\mathrm{E}[E_i(p)] = \sigma^2_{E_i(p)}$. Since the image is a flat field, the expected value of the exposure can be approximated using the spatial variance, i.e., $\mathrm{E}[E_i(p)] \approx \mathbb{V}\mathrm{ar}_{p\in\mathcal{I}}[\mathit{ff}(p)]$. However, this variance estimate includes not only the shot noise of the exposure, but also the readout noise and the PRNU.

Janesick observes that the PRNU can be virtually eliminated by taking the difference of two flat fields. The spatial variance of the difference between two flat fields $\mathit{ff}(p), \mathit{ff}'(p)$ can be approximated as

$$\begin{aligned}
\underset{p\in\mathcal{I}}{\mathbb{V}\mathrm{ar}}[\mathit{ff}(p) - \mathit{ff}'(p)] &= 2\,\underset{p\in\mathcal{I}}{\mathbb{V}\mathrm{ar}}[\mathit{ff}(p)] \\
&\approx 2\,\underset{p\in\mathcal{I}}{\mathbb{V}\mathrm{ar}}[\mathrm{gt}\,(\mathrm{a}(p)X_i(p) + D_i(p)) + N_R] \\
&= 2\left[\mathrm{g}^2\left(\underset{p\in\mathcal{I}}{\mathbb{V}\mathrm{ar}}[\mathrm{ta}(p)X_i(p)] + \underset{p\in\mathcal{I}}{\mathbb{V}\mathrm{ar}}[\mathrm{t}D_i(p)]\right) + \sigma_R^2\right] \\
&= 2\left[\mathrm{g}^2\mathrm{t}\left(\mu_X(1+\sigma_\mathrm{a}^2) + \mu_D\right) + \sigma_R^2\right],
\end{aligned} \tag{2.32}$$

where σ_a^2 is the PRNU spatial variance. Here, it is assumed that the dark current has the same expected value across the image domain. Additionally, the expected value for a single flat field can be approximated as

$$\begin{aligned}
\underset{p\in\mathcal{I}}{\mathbb{A}\mathrm{vg}}[\mathit{ff}(p)] &\approx \underset{p\in\mathcal{I}}{\mathbb{A}\mathrm{vg}}[\mathrm{gt}\,(\mathrm{a}(p)X_i(p) + D_i(p)) + N_R] \\
&= \mathrm{g}\,\underset{p\in\mathcal{I}}{\mathbb{A}\mathrm{vg}}[\mathrm{ta}(p)X_i(p) + \mathrm{t}D_i(p)] + \mathrm{L}_0 \\
&= \mathrm{gt}(\mu_X + \mu_D) + \mathrm{L}_0.
\end{aligned} \tag{2.33}$$

Therefore, by combining Eq. 2.32 and Eq. 2.33, the gain factor can be esti-

mated as

$$\hat{\mathrm{g}} = \frac{1}{k} \cdot \frac{\frac{1}{2}\,\mathbb{V}\mathrm{ar}_{p\in\mathcal{I}}[ff(p) - ff'(p)] - \sigma_R^2}{\mathbb{A}\mathrm{vg}_{p\in\mathcal{I}}[ff(p)] - \mathrm{L}_0}, \text{ where} \tag{2.34}$$

$$k = \frac{\mu_X(1+\sigma_{\mathrm{a}}^2) + \mu_D}{\mu_X + \mu_D}. \tag{2.35}$$

Since σ_a^2 is usually very low ($\sigma_a \approx 1\%$), the factor k is usually omitted in the literature. Lastly, several gain estimates should be averaged in order to reduce the influence of readout noise; these estimates should be obtained from the difference between pairs of distinct flat fields.

PART I

Editing of Exposure Sequences for HDR Imaging

Whenever we take a photograph of a scene, it is possible that the light in the scene exceeds the dynamic range of that the camera can capture. Without altering the camera's hardware, this limitation can be overcome by acquiring sequences of images with different exposure time known as *exposure sequences*. These images are averaged into a single image with higher dynamic range, an *HDR image*, that contains the light intensities measured in every image. This is possible whenever the scene is static, or if the motion in the scene can be compensated. Otherwise, when the scene contains dynamic objects whose motion cannot be determined, these objects need to be detected and excluded from the average to avoid introducing *ghosting artifacts* in the final image.

Chapter 3 shows how the camera noise model (described in Sec. 2.2.3) provides valuable visual clues to detect and remove moving objects from exposure sequences. These clues, together with additional plausibility constraints, are encoded in a global energy functional that is minimized using graph cuts. In this way, HDR images can be produced without any ghosting artifacts. Chapter 4 presents three other applications of the camera noise model in the context of HDR image processing; these are: HDR image reconstruction with optimal signal-to-noise ratio (SNR), estimation of optimal exposure times for producing HDR images with maximum SNR, and de-noising of HDR images.

CHAPTER 3

Noise-aware HDRI Deghosting

3.1 Introduction

The range of light intensity that cameras can measure in a single exposure is known as its *dynamic range*. This range is limited in current digital cameras: In current CCD and CMOS sensors, each pixel has a capacitor to store incident photon-electrons. Each capacitor can store a limited amount of charge before it reaches *full-well capacity* or *saturation*. When saturation occurs, it is visible in the final image as *over-exposure* artifacts. In addition, the camera is affected by several noise sources during the conversion from charge to digital values. These sources are known as *readout noise*. It destroys the signal of low photo-electron measurements, which occur on dim scenes or short exposures. When this occurs, it is visible as *under-exposure* artifacts in the resulting images. The dynamic range of digital cameras is limited by this incapability to measure very dim and very bright light intensities in a single image.

Images that contain a larger dynamic range are known as *high dynamic range (HDR)* images. Recording such HDR images is a relevant problem, since many applications require measuring the complete dynamic range of a scene. For instance, the realism of rendered images can be increased by reproducing the complex lighting present in natural scenes. This can be achieved by capturing environment maps (360 degrees HDR images) and use them to simulate natural, distant lighting, in a process called *image-based lighting* [Debevec98a]. Using realistic materials can also improve the realism of renderings: The reflectance properties of a real world materials can be

measured from sets of HDR images taken from different viewpoints and with different incident lights, in a process called *bi-directional reflectance function (BRDF) reconstruction* [Lensch03]. In photography, artists also benefit from the capability of capturing the extended dynamic range of a scene. This is reflected in the proliferation of software tools for creating high HDR images from sets of photographs. In particular, effects such as simulated motion blur become more realistic if HDR images are used [Debevec97].

The main strategy for capturing HDR images is to average multiple images of different exposure time. If each image samples a different intensity range, the combined result will have a larger dynamic range than any of them. For instance, in a room with sunlight coming through a window, short exposures can properly measure bright objects that reflect sunlight, whereas long ones can capture objects sitting in darker places. A single image is not sufficient since bright objects saturate the sensor in long exposures, and dark objects are not visible in the short exposures due to readout noise. A HDR image can be produced from a weighted average of these images following methods known as *HDR reconstruction* and *image fusion*.

Methods for HDR reconstruction [Mann95, Debevec97, Robertson03, Mitsunaga99, Mann01, Tsin01, Reinhard05b, Granados10, Hasinoff10] produce scalar images whose values are proportional to the irradiance in the scene. They achieve this by converting the digital images back to the irradiance domain before performing the average. If the exposure time t_i, and the mapping $\mathrm{f} : \mathbb{R} \to \mathbb{Z}^+$ from irradiance to digital values or *camera response* (see Eq. 2.18) are known, the irradiance $X_i(p)$ measured by the i-th image at pixel p can be estimated as

$$X_i(p) \approx \frac{\mathrm{f}^{-1}\left(V_i(p)\right)}{\mathrm{t}_i}, \tag{3.1}$$

whenever the observed pixel value is not saturated, i.e., $V_i(p) < \mathrm{L}_{sat}$ (see Sec. 2.2.4).

An *irradiance map* of the scene contains the non-saturated observations in every image i. This irradiance map X is estimated as the weighted average

$$\hat{\mu}_{X(p)} = \frac{\sum_{i \in T} W_i(p) X_i(p)}{\sum_{i \in T} W_i(p)}, \tag{3.2}$$

where $W_i(p)$ corresponds to the weight for the i-th image on pixel p, and $T = \{1, \ldots, n\}$ is the index set of the image sequence.

Irradiance maps can no longer be visualized in standard low dynamic range (LDR) devices, such as monitors or printed media, due to their larger dynamic range. For displaying them in LDR devices, their range needs to be clipped or compressed; this process is known as *tone mapping* [Larson97].

Image fusion [Burt93, Mertens09] methods are more suitable whenever the objective is to display the HDR images in LDR displays. These methods

average the digital images directly, by selecting the best exposure for every object, e.g. bright objects from short exposures, dark objects from long exposures. In this way, fusion methods avoid the conversion between digital values and irradiance values that is required if HDR reconstruction followed by tone mapping is performed.

3.2 HDR Deghosting

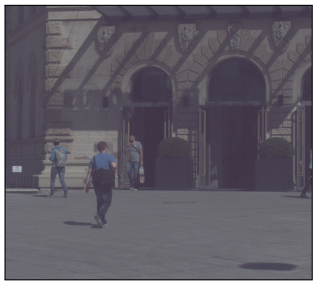
HDR reconstruction and image fusion methods are only valid for static scenes, since when averaging different images one implicitly assumes that the camera and the objects in the scene remain static. Whenever this condition is not satisfied, averaging images of different objects introduces *ghosting* artifacts in the result, as illustrated in Fig. 3.1. In previous work, this limitation has been addressed either by aligning the moving objects before the averaging, or by detecting regions with moving objects and excluding their images from the average. These two strategies are explained in the following sections.

3.2.1 Motion-compensation Methods

Following the first strategy, motion compensation, Bogoni [Bogoni00], Kang et al. [Kang03], and Zimmer et al. [Zimmer11] reduce ghosting artifacts by performing a dense alignment of the images prior to averaging using optical flow. They compute the optical flow in the gradient domain in order to cope with color discrepancies caused by the differences in exposure time. Although optical flow methods can correct short displacements caused by camera shake and moving objects, they often fail to estimate large displacements of small objects, such as people moving in the scene. Therefore, the success of deghosting methods based on motion compensation depends on the accuracy of the estimated flow field, so in many cases this strategy fails (see Sec. 3.8.4).

3.2.2 Detection-and-exclusion Methods

Most HDR deghosting methods follow a detection-and-exclusion approach. They can be classified into three categories according to the exclusion strategy they follow: *a*) Methods that detect ghosted regions and select a single image (or the average of compatible images) to represent them, *b*) methods that use a reference image and average only similar colors, and *c*) methods that build a reference background model and average only the compatible colors. These strategies are described next.



(a) $t_1 = \frac{1}{2500}s$

(b) $t_2 = \frac{1}{1000}s$

(c) $t_3 = \frac{1}{250}s$

(d) HDR image with ghosting artifacts

Figure 3.1: Example of ghosting artifacts when combining images that contain moving objects: (a)-(c) input LDR images; (d) Tone-mapped irradiance map, with under-, over-exposed values excluded from the average.

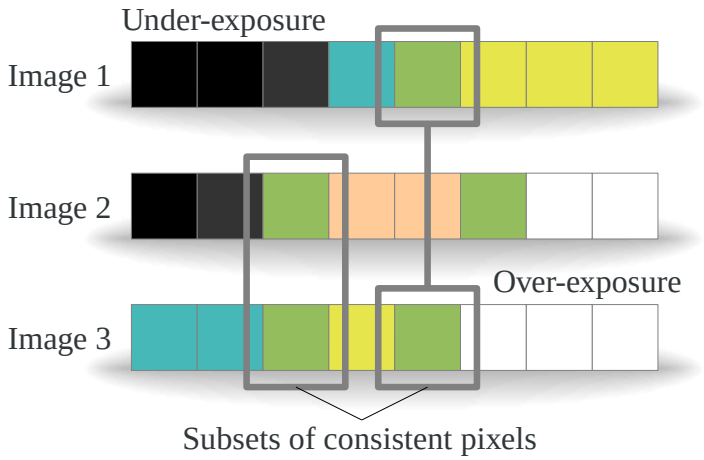


Figure 3.2: 1-D illustration of ghost-free HDR reconstruction by detecting subsets of consistent pixel values. An HDR image can be reconstructed by averaging the irradiance estimates derived from the color of corresponding pixel location in the input images. Ghosting artifacts appear whenever sets of inconsistent colors are included in the average. The problem of HDR deghosting can be defined as selecting consistent subsets of colors for every pixel.

Detection-and-selection Methods

The first type of methods classifies pixel locations into *consistent* and *inconsistent*. A pixel is consistent across a set of images when the observed color values correspond to the same incident light to the sensor, i.e., they show the same object. The final color value of consistent pixels is obtained as an average of the observed pixel values at that location on all images, whereas the color value of an inconsistent pixel is obtained by *choosing* a single well-exposed image or a consistent subset of images (see Fig. 3.2).

The seminal method by Ward [Reinhard05b] detects regions where the variance of the irradiance estimates is higher than a fixed threshold. For each detected region, they chose a single well-exposed image as representative. In addition, Jacobs et al. [Jacobs08] observe that the variance is not high on low-contrast regions that undergo local motions, such as trees or water. To handle such cases, they find regions where the difference in the spatial color variation is large, i.e., the difference between the entropy of the empirical distribution of color values is high.

Grosch [Grosch06] uses a simpler detector: A region is marked as ghosted if the absolute irradiance difference between any pair of images exceeds a given threshold.

Following a different strategy, Sidibé et al. [Sidibé09] detect regions where the relation of monotonic increase between exposure time and pixel

values is broken. In addition to this criteria, Silk and Lang [Silk12] detect and exclude pairs of regions where the relative irradiance difference is larger than a threshold.

Lastly, under the assumption that the median value of every image of a scene is invariant to the exposure time [Ward03], Pece and Kautz [Pece10] detect regions where the ordering of observed values with respect the median is not consistent on all images.

However, the quality of the HDR images produced by these methods depends on the precision and recall rate of the motion detection. False-positives will produce ghosting artifacts in the result, whereas true-negatives reduce the signal-to-noise ratio (SNR), as fewer images are averaged. The effect of each of these types of errors in the ghosting prevention and in the SNR is also illustrated in Fig. 3.6, where either false-positives or true-negatives are obtained according to the accuracy of the ghosting detection method.

Reference-based Methods

A second group of methods excludes from the average those observations that are not compatible with a predefined reference image. This strategy can avoid ghosting artifacts since each LDR image is self-consistent by definition [Reinhard05b].

Menzel and Guthe [Menzel07] average regions whose cross correlation with the reference in a small neighborhood or *patch* falls below a threshold. Gallo et al. [Gallo09] includes a patch in the average only if the majority of its irradiance values can be mapped linearly to the reference image. Following the same strategy, Raman et al. [Raman10] avoid transforming the images to the irradiance domain by fitting a polynomial function to the mapping between corresponding intensities in a candidate and a reference patch. This function is known as the *intensity mapping function (IMF)*. Closely related, Raman and Chaudhuri [Raman10] estimate the IMF using only regions of low variance across images. Heo et al. [Heo10] follow a similar strategy where the probability that an observation satisfies the IMF is approximated by the histogram of corresponding intensities between the reference and the other images.

On the other hand, the method of Min et al. [Min09] assumes that the histogram of the reference image and the remaining images is similar. They select those observations where the bin index matches that of the reference. Park et al. [Park11] address the simpler case where only two images are available; whenever the intensity difference exceeds a threshold the shorter exposure is preferred. Last, Zhang and Cham [Zhang12b] select images where the gradient orientation is similar to the reference.

Following a different strategy, Sen et al. [Sen12] perform local image alignment in addition to reference-based HDR reconstruction. Their method

can handle camera and scene motion by defining a reference image to which other images are patch-wise aligned [Barnes09]. Ill-exposed regions in the reference are filled using an adaptation of the bi-directional similarity function [Simakov08] between the remaining input images and the HDR result.

However, the main drawback of this category of methods is that the reference is itself an LDR image. LDR images are in general incomplete since some regions will be under- or over-exposed if the dynamic range of the scene exceeds that of the camera. As there the reference is undefined for such regions, it is possible that incompatible images be selected to fill them, which could easily lead to inconsistent HDR images.

Model-based Methods

A third strategy is to build a model of the static part of the scene or *background* to guide the averaging process. The distance between every observation and the background can be used to reduce the weight of inconsistent observations in the average. For building a background model, it is assumed that the background is observed more frequently than other moving objects.

Khan et al. [Khan06] builds a background model that estimates the irradiance probability distribution at every pixel. The model is produced using kernel density estimation. In the average, they weight observations according to their estimated probability. Pedone and Heikkilä [Pedone08] extend this approach by performing automatic bandwidth estimation, assuming that the observations belong to the same distribution, which is valid only if the scene is static. Granados et al. [Granados08] construct a background model by selecting a single representative image per pixel. For each pixel, the representative images are selected according to their probability. Using the representative as reference, they reconstruct the final HDR image by averaging only those images that are closer than a threshold.

Following a similar strategy, Tomaszewska and Markowski [Tomaszewska10] use the average irradiance as background model, and down-weight observations that are far from the mean. Similarly, Zhang and Cham [Zhang12a] propose using the average gradient orientation as background model.

For these methods, the consistency of the resulting HDR image depends only on the accuracy of the estimated model, assuming a dominant background. Additionally, constructing a background model requires having a larger number of well-exposed observations. However, the assumption of dominant background is easily broken in scenes with several moving objects (e.g. a street with many pedestrians), and the number of observations is reduced by under- or over-exposures. Therefore, the applicability of model-based methods is more limited in practice.

3.3 Proposed Uncertainty-based Method

The previous section discussed several HDR deghosting methods. However, HDR deghosting still remains an open problem: A recent report by Srikantha and Sidibé [Srikantha12] compares several state-of-the-art methods and concludes that "there is no single best method and the selection of an approach depends on the user's goal".

As explained in the previous section, deghosting methods based on ghosting detection and exclusion (Sec. 3.2.2) rely on the ability to test if the colors observed on the same pixel on different images are inconsistent. These methods find such consistent pixels based on several criteria such as irradiance difference between observations [Grosch06, Park11, Silk12], irradiance difference to a reference image [Grosch06, Granados08], distance to the intensity mapping function [Gallo09, Raman10, Raman10], sum of squared differences to the estimated irradiance (variance) [Reinhard05b, Jacobs08], average ratio between between images [Tomaszewska10], probability of the distance to a background model [Khan06, Pedone08], correlation with a reference image [Menzel07], difference of the entropy on local image patches [Jacobs08], and difference between gradient orientations [Zhang12b, Zhang12a]. Each of these consistency tests requires setting fixed thresholds that are unlikely to generalize well to the noise properties of different cameras and scenes. Other strategies such as color quantization and bin matching [Min09, Pece10], and tests for monotonic intensity increase with exposure [Sidibé09] can be cast as alternatives for dealing with detection problems caused by differences in exposure and noise. These alternatives test invariants that have high specificity but have lower sensitivity than other methods (see Sec. 3.8.5).

In this chapter, I claim that the reliability and working range of HDR deghosting can be significantly improved by modeling the noise distribution of the color values measured by the camera: In order to test if two colors observed at the same pixel location in different images correspond to same irradiance, it is necessary to take into account their noise distributions. This claim is experimentally verified in several scene conditions.

The noise distribution of the input images has been largely neglected in previous work. This distribution depends on the camera and exposure settings (see Fig. 3.3), and it can be modeled using a Gaussian distribution, where its variance is proportional to the light intensity, inversely proportional to the squared exposure time, and depends on camera parameters such as the gain factor, and the readout noise parameters (Sec. 3.4). Given that the noise depends on the scene irradiance and the camera parameters, it can be expected that no fixed threshold can be set to reliably detect image differences across camera models and scenes. Following this observation, I propose to normalize the consistency tests using the predicted noise distribution of the input images (Sec. 3.5 and 3.6). In general, there can be

a.	Take an input set of images with a static camera.
b.	If not provided by the manufacturer, estimate the readout noise using an additional black frame, and the camera gain using the input images (Sec. 3.4).
c.	Select a consistent subset of images for every pixel (Sec. 3.5 and 3.6).
d.	Reconstruct the irradiance of each pixel from an plausible arrangement of consistent sets (Sec. 3.7).

Table 3.1: Summary of the proposed HDR deghosting pipeline.

multiple ghost-free HDR images that are consistent with the given set of input images. Among them, the final HDR image is constructed such that each pixel has high signal-to-noise (SNR) ratio and is spatially compatible to its neighbors (Sec. 3.7).

The resulting algorithm is the first HDR reconstruction method to handle scenes with strong clutter and dynamics without introducing ghosting artifacts. This type of scene complexity has not been addressed in the existing literature. The proposed method also performs on par with state-of-the-art methods on image sets that show only small object displacements. Furthermore, the novel use of a camera noise model allows it to produce results with lower noise than other methods, even on images acquired at low light such as night shots (Sec. 3.8). The proposed pipeline is summarized in Table 3.1. The resulting method was published in [Granados13].

3.4 Image Variance Derivation

Even assuming a static scene and constant camera parameters, the variance of the irradiance estimates can change from image to image. This occurs as the noise distribution varies significantly with the exposure time, in addition to the differences in the amount of light collected by the sensor. The noise distribution can be estimated for the *raw output* of the camera, i.e., the output before performing operations such as demosaicing, white balancing, intensity mapping, color enhancement, sharpening, and compression.

For the problem of deghosting, it is only necessary to consider the temporal noise sources affecting the images (described in Sec. 2.2.1). This is justified as the ghosting detection is performed for each pixel independently, i.e., if the detection evaluates color differences independently, variations in the noise distribution *between* pixel locations caused by spatial noise sources are not relevant to the detection.

The two temporal noise sources that affect the image formation process are *shot noise*, and *readout noise*. Shot noise is caused by the process of light emission, which is well modeled by a Poisson distribution, where the

variance is equal to the mean. Readout noise comprises several other signal-independent sources affecting the acquisition process in digital cameras. It is well modeled by a Gaussian distribution with zero mean. The addition of these two noise sources can be approximated using a Gaussian distribution, for high enough light intensities.

For a given image, the amount of observed photon-electrons and its variance can be estimated if the inverse of the camera response function is known (see Sec. 2.2.3). For raw camera output, the inverse function can be represented as

$$\mathrm{f}^{-1}(V_i(p)) \approx \frac{V_i(p) - B_i(p)}{\mathrm{g}}, \tag{3.3}$$

where B is a dark current image (i.e., an image with same exposure time as V, but no incoming light), g is the gain factor that depends of the camera ISO setting. The proposed method assumes that the dark current $B_i(p)$ is negligible, or equivalently, that the dark frame subtraction is performed in-camera. Therefore, the dark current is replaced with the offset of the camera output L_0.

From Eq. 2.20 and Eq. 3.3, the variance of the an irradiance estimate $X_i(p)$ can be approximated by

$$\sigma^2_{X_i(p)} = \frac{\mathrm{g}^2 \mathrm{t}_i X_i(p) + \sigma_R^2}{\mathrm{g}^2 \mathrm{t}_i^2}. \tag{3.4}$$

where σ_R^2 is the variance of the readout noise. For illustration, Fig. 3.3 shows the differences in noise between images of the same scene taken with different exposure time. Note that on static scenes, the image variance $\sigma^2_{X_i(p)}$ in Eq. 3.4 also could be estimated from a large sample of images with the same exposure time. However, HDR reconstruction requires to capture only as many images required to cover the full dynamic range of the scene (three images is a common choice), which is not sufficient to estimate the image variance. In addition, it is less practical to capture a large set of images for reconstructing the dynamic range of a single scene, and it is often impossible on scenes where moving objects are unavoidable. For these reasons, the proposed deghosting method uses a camera noise model (see Sec. 2.2) to predict the variance of every image.

For evaluating Eq. 2.23 and Eq. 3.4, the parameters g, L_0, σ_R^2, and t_i need to be estimated. The exposure time t_i can be read directly from the digital image file. The estimation procedure of the remaining parameters is explained next.

3.4.1 Readout noise

The black level L_0 and the readout variance σ_R^2 can be calibrated using the method described in Sec. 2.2.4. This method estimates L_0 and σ_R^2 as the

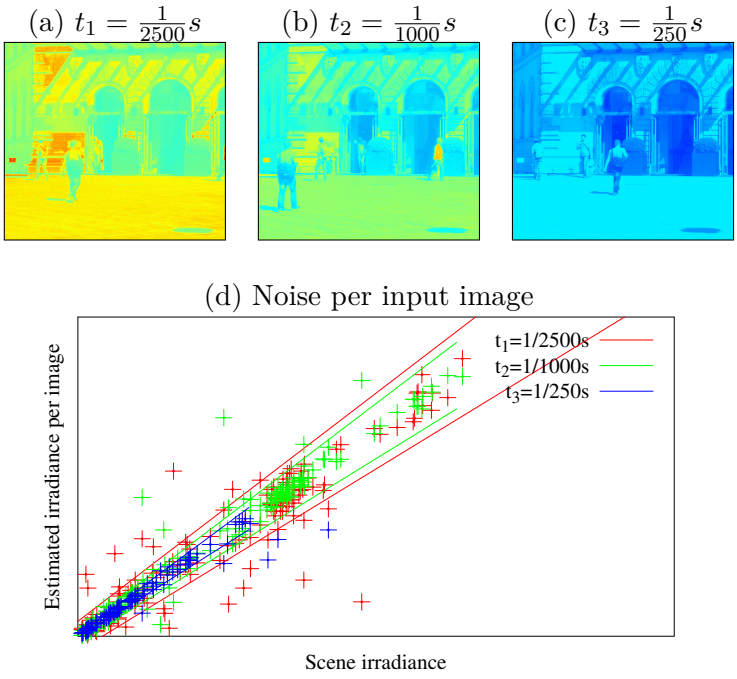
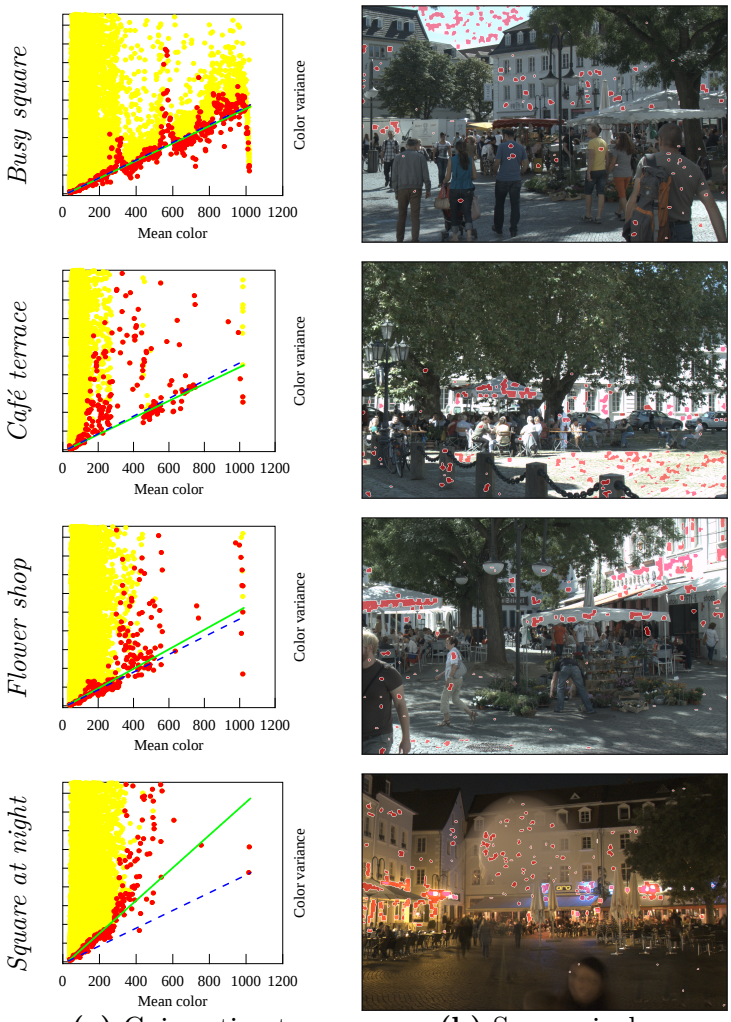


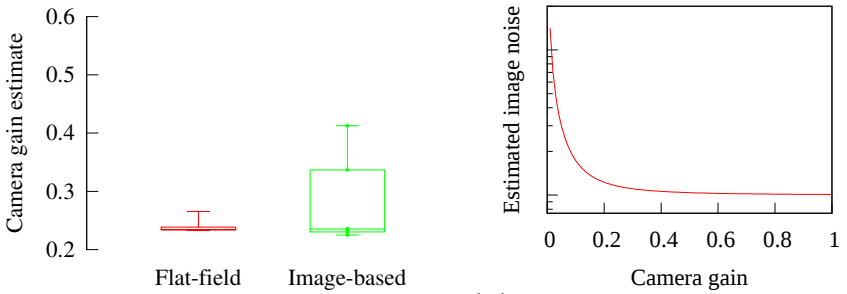
Figure 3.3: Estimated uncertainty for the three images of the scene shown in Fig. 3.1. These images have a exposure time difference of two stops. (a)-(c) Per-pixel standard deviation $\sigma_{X_i(p)}$ of the irradiance estimations of each image; blue and red represent low and high standard deviations, respectively; (d) confidence interval of 99.5% for each image. Note that the interval varies for the exposure times and irradiance intensities, and that samples that correspond to moving objects fall outside the confidence interval.

mean and variance, respectively, of the pixel values of a *black frame*, i.e., an image taken with no incident light and no integration time (or very short exposure time). For the experiments presented in this chapter, the readout noise calibration was performed manually from black frames, but in practice this data could be provided for every camera model by the manufacturer.



(a) Gain estimate **(b)** Super pixels

Figure 3.4: Image-based gain calibration. **(a)** The mean and variance of each super pixel are shown in a scatter plot, where low-variance super pixels are shown in red, and the remaining (high-variance) super pixels are shown in yellow. In these plots, the green lines show the predicted color variance using image-based calibration, whereas blue dashed lines show the results using flat-field calibration. **(b)** The red regions correspond to super pixels with low-color variance found in one of the input images.



(a) Gain confidence by calibration method

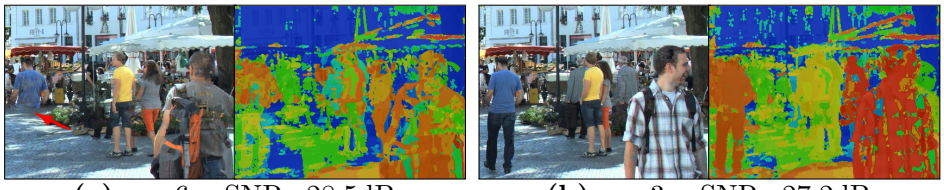
(b) Effect of calibration error on image noise estimation

Figure 3.5: Confidence of camera gain estimation. **(a)** The 1st, 25th, 50th, 75th and 99th percentiles of flat-field calibration (sample of 36 flat field images), and image-based calibration (sample of seven images, each from a different scene; two shown in Fig. 3.4). The expected gain for both methods are very close, but the variance of image-based calibration is higher. Despite its higher variance, our gain estimate can be used to reconstruct ghost-free HDR images (see Fig. 3.6). **(b)** In general, when the camera gain is over-estimated, the predicted noise for the input images is under-estimated. This makes the ghosting detection stricter, thus reduces the SNR of the final HDR image (because smaller consistent subsets will be found). Still, no ghosting artifacts are introduced (see Fig. 3.6).

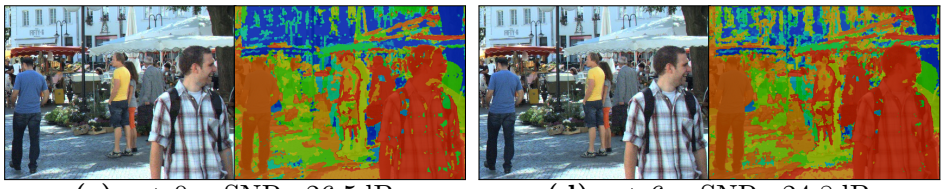
3.4.2 Camera gain

The camera gain g can be accurately calibrated using *flat fields*, i.e., images exposed with a constant illumination at every pixel, such that every pixel color can be assumed to be a sample of the same random variable (see Sec. 2.2.4). Under this assumption, the mean and variance of the observed color can be approximated using the spatial mean and variance of a flat field. Using this approximation, the gain can be derived by exploiting the equivalence between the expected value and the variance of the exposure. This *flat-field calibration* is the best method available, and it can be applied to any digital camera. However, in practice, this requires additional flat field images, which may be cumbersome for inexperienced users to acquire.

To overcome this limitation, I propose an alternative *image-based calibration* that does not require acquiring flat fields and that works directly off the input images of the given scene. The underlying idea is to use regions of constant illumination in the input images as proxies for the flat fields. To facilitate this process, the input image (e.g. the best exposure of the input set) is segmented into *super pixels* [Veksler10], and the mean and variance of their color values is estimated. From the resulting mean-variance scatter plot (Fig. 3.4a), the minimum variance is selected for each digital value, and RANSAC [Fischler81] is applied for fitting a line that passes through



(a) g $- 6\sigma$, SNR=28.5dB **(b)** g $- 3\sigma$, SNR=27.2dB



(c) g $+ 0\sigma$, SNR=26.5dB **(d)** g $+ 6\sigma$, SNR=24.8dB

(e) g $+ 12\sigma$, SNR=23.6dB **(f)** g $+ 24\sigma$, SNR=21.1dB

Figure 3.6: Robustness of the deghosting algorithm to the inaccuracy of the camera gain calibration. Each pair of images corresponds to the tone-mapped HDR reconstruction (left) and its corresponding labeling (right) for a specific level of calibration error. The SNR values are computed as the average ratio of the estimated irradiance and the standard deviation, i.e. $20\,\mathbb{A}\text{vg}[\log_{10}\hat{\mu}_{X(p)}/\hat{\sigma}_{X(p)}]$. Our algorithm is robust against slight under-estimation and large over-estimation of the camera gain parameter. When the gain is under-estimated, the predicted image noise becomes over-estimated. This makes the irradiance equality test more lenient, thus leading to ghosting artifacts (a, red arrow). However, this case was never observed in our experiments since the color variance (on which it depends) is never under-estimated. On the other hand, when the gain is over-estimated, the image variance is under-estimated, which leads to more strict consistency tests. This lowers the signal-to-ratio (SNR) of the result (as the algorithm finds smaller consistent subsets of images), but does not cause ghosting artifacts (c-e). Even when over-estimation of the gain parameter occurs in practice, our algorithm still creates plausible HDR images.

$(\mathrm{L}_0, \sigma_R^2)$, i.e., through the expected variance at the black level. Figure 3.4 illustrates this process: Figure 3.4a shows the mean and variance color value of each super pixel (yellow and red dots). Among them, the super pixels with minimum variance are selected as the proxies for flat fields (shown in red in Fig. 3.4a and 3.4b). This selection is justified as only shot noise and readout noise contribute to the variance of image regions with constant illumination, and therefore, these noise sources determine the lower bound of the color variance. Using super pixels for estimating the lower bound of the variance of images has been previously proposed in [Liu08] for image denoising.

Figure 3.5 demonstrates the performance of each gain estimation method: The proposed image-based calibration is sufficiently accurate, and it is comparable with the flat-field calibration. In addition, since a wide range of scenes contain locally flat regions, the proposed deghosting algorithm can be directly applied to them without requiring users to provide additional flat field images. On the other hand, Fig. 3.4–bottom (i.e. *square at night* scene) provides an example image from which the camera gain could not be correctly estimated. This image contains flat regions but they cover only a very limited color band, which misleads the slope estimation (Fig. 3.4a–bottom). Nevertheless, the user can be sloppy about the calibration, but the deghosting algorithm tolerates even large inaccuracies in the gain calibration, which shows its practical applicability by un-experienced users. The price to be paid for the errors in automatic gain calibration is the degradation in SNR. However, this degradation is graceful with increasing calibration error, without introducing ghosting artifacts (see Fig. 3.6 for the effect of calibration errors on the final result).

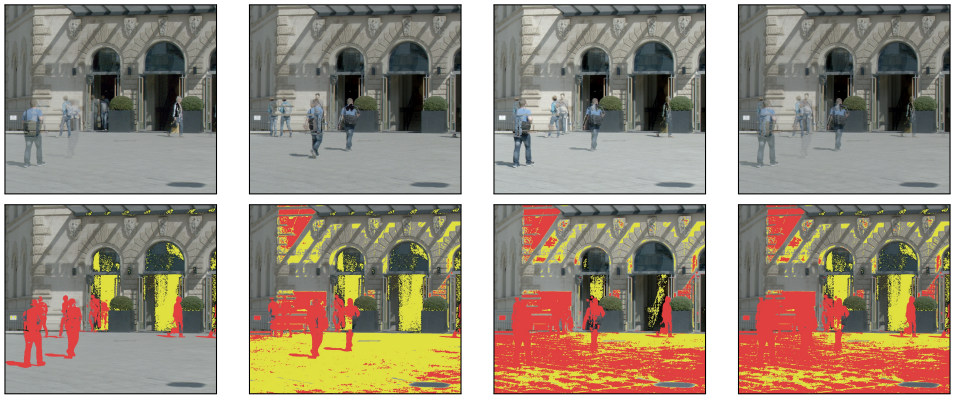
Lastly, gain calibration needs to be done only once per camera model, and therefore, this parameter can be also provided by the manufacturer. Alternatively, a public database of calibrated parameters could be shared among users. The results presented in this chapter are computed using the proposed image-based calibration procedure on each sequence.

3.5 Consistency Test for Pairs of Images

Let us assume that two irradiance observations $X_i^k(p)$, $X_j^k(p)$ are given at pixel p and color channel k, which are derived from the pixel colors $v_i^k(p)$, $v_j^k(p)$ on images i, j, respectively, using the inverse of the camera response (Eq. 2.23). The task of detecting ghosting artifacts boils down to testing if these irradiance observations are *consistent*, i.e., if they correspond to measurements of the same incident light. Existing algorithms solve this problem by relying on pre-determined thresholds, which are unlikely to generalize well to different cameras and scenes. This requirement can be avoided by exploiting the image noise estimation obtained in Sec. 3.4.

(a) (b) (c) (d) (e)

Figure 3.7: Computation of the consistency probability between two images of the same scene with different exposure time: (a) Absolute difference between the images t_1, t_2 shown in Fig. 3.1; (b) noise-normalized difference, which makes the differences intensity independent; (c) consistency probability, which is also intensity independent; (d) reconstructed irradiance map from the two images; (e) overlay with consistency mask obtaining by thresholding the consistency probability with some value α; inconsistent regions are shown in red. An example of the result of ghost detection using absolute differences is shown in Fig. 3.23e.



(a) $\{t_1, t_2\}$ (b) $\{t_1, t_3\}$ (c) $\{t_2, t_3\}$ (d) $\{t_1, t_2, t_3\}$

Figure 3.8: Thresholded consistency probability for different subsets of the images shown in Fig. 3.1: (top) Irradiance map estimated from each subset; (bottom) overlay with the corresponding inconsistency mask (red) and ill-exposure mask (yellow). The inconsistency mask is obtained by thresholding the consistency probability using a fixed confidence value α. All pixels not marked in red are considered as consistent. Note that the ill-exposed pixels do not cause subsets to be immediately marked as inconsistent (see Sec. 3.7.1) but they are excluded from the irradiance average (Eq. 3.11).

The proposed approach is to estimate the probability distribution of a difference function $d_{ij}^k(p) = X_i^k(p) - X_j^k(p)$: Since $X_i^k(p)$ and $X_j^k(p)$ are Gaussian, $d_{ij}^k(p)$ is also Gaussian which for consistent pairs has mean zero and variance

$$\sigma^2_{d_{ij}^k(p)} = \sigma^2_{X_i^k(p)} + \sigma^2_{X_j^k(p)}, \tag{3.5}$$

where $\sigma^2_{X_i^k(p)}$ and $\sigma^2_{X_i^k(p)}$ are obtained from Eq. 3.4. Given the variance $\sigma^2_{d_{ij}^k(p)}$, the *consistency probability* that observations at pixel p on images i, j are consistent is estimated by comparing the corresponding irradiance *differences* with the expected noise distribution of the images on every color channel:

$$\Pr\left(p\,|\{i,j\}\right) = \min_{k\in\{R,G,B\}} \Pr\left(-\frac{|d_{ij}^k(p)|}{\sigma_{d_{ij}^k(p)}} \leq \mathcal{N} \leq \frac{|d_{ij}^k(p)|}{\sigma_{d_{ij}^k(p)}}\right), \tag{3.6}$$

where $\mathcal{N}$ is the standard Gaussian random variable with mean zero and variance one. In practice, the estimate $\Pr\left(p\,|\{i,j\}\right)$ can be very noisy (e.g. when the image is taken under low-light or when the camera has a high read-out noise). For this reason, prior to estimating the probabilities, difference image $d_{ij}^k(p)$ is smoothed by applying bilateral filtering [Tomasi98] using a distance kernel with fixed large bandwidth (13 pixels) and a range kernel with variable bandwidth $\sigma_r = 2\sigma_{d_{ij}^k(p)}$ that is proportional to the expected image noise. The resulting probabilities are illustrated in Fig. 3.7.

It should be noted that since the noise variance σ_x^2 is different at every pixel and image in the sequence, the variance of the difference function $\sigma^2_{d_{ij}^k(p)}$ also varies for every pixel and image pair. This suggests that when the noise variances are not properly taken into account, the existing threshold-based approaches are not likely to generalize well to the noise properties of different cameras and exposure settings.

3.6 Consistency Test for Sets of Images

Let $\mathbf{V} = \{v_i\}_{i\in T}$ be the set of images in the exposure sequence. Based on the pair-wise consistency measure (Eq. 3.6), the probability that a given subset $S_l \in 2^{\mathbf{V}}$ is consistent at a pixel p is defined as the minimum of the pair-wise consistency:

$$\Pr\left(p\,|S_l\right) = \min\left\{\Pr\left(p\,|\{i,j\}\right)\right\}_{\{i,j\}\in S_l\times S_l}. \tag{3.7}$$

An example of the resulting consistency probabilities is shown in Fig. 3.8.

For the case of a singleton S_l (i.e., $|S_l| = 1$) the corresponding consistency probability is given as the probability that the corresponding observation is

well-exposed:

$$\Pr(p|\,\{i\}) = 1 - \max\left\{\min_k \Pr_{\text{ue}}(v_i^k(p)), \max_k \Pr_{\text{oe}}(v_i^k(p))\right\}, \tag{3.8}$$

with $k \in \{R, G, B\}$. $\Pr_{\text{ue}}$ and $\Pr_{\text{oe}}$ correspond to the under- and over-exposure probability, respectively, of a single observation according to the distribution of the readout noise. In this definition, *all* color channels need to be under-exposed for considering an observation $v_i(p)$ as inconsistent, whereas *any* over-exposed color channel renders it as inconsistent.

3.7 Compositing of consistent sets

For obtaining a ghost-free HDR image, a consistent subset of irradiance estimates is selected for every pixel to be used for reconstructing the final pixel value. However, given the presence of moving objects, there can be more than one consistent subset. Arbitrarily selecting any one of them may introduce unnatural color discontinuity in the final image (see yellow arrows in Fig. 3.9). We resolve this problem by introducing a spatial continuity measure as a regularizer. The resulting algorithm is a global energy minimization framework that takes into account the consistency at every pixel location as well as their spatial coherence. Our final result is represented as a labeling $F_p := F(p)$ that assigns to each pixel p the index of an element in $2^{\mathbf{V}}$. This labeling is obtained by minimizing the energy functional

$$\begin{aligned}\mathcal{E}(F) = \quad & \sum_{p\in\Omega} \Big(\underbrace{\mathbb{1}_{\{\Pr(p\,|S(s)_{F_p})>\alpha\}}}_{\text{consistency potential}} + \underbrace{\gamma V(S(s)_{F_p})}_{\text{variance potential}} \Big) + \\ & \beta \sum_{(p,q)\in\mathcal{N}} \underbrace{\mathbb{1}_{\{\Pr(p\,|S(s)_{pq})>\alpha \vee \Pr(q\,|S(s)_{pq})>\alpha\}}}_{\text{prior potential}},\end{aligned} \tag{3.9}$$

where $S(s)_{pq}$ corresponds to the index of the subset $S(s)_{F_p} \cup S(s)_{F_q} \in S(s)$, $\mathcal{N}$ denotes the 4-neighborhood system, and β and γ are hyper-parameters.

In Eq. 3.9, the roles of the *consistency potential* and the *variance potential* are to ensure that the final reconstruction is consistent at every pixel, and that it is not noisy, respectively (as discussed shortly). The role of the *prior potential* is to encourage that the final reconstruction at every pixel agrees with its spatial neighbors. In addition, instead of penalizing the consistency probability directly, a confidence value α is defined to determine whether a set of images $S(s)_{F_p}$ is consistent or not. This encodes an important design choice: The aim is to select *any* consistent group, not the *most* consistent one. This design gives more freedom to the optimization algorithm in constructing the final composite.

In Sec. 3.5, well-exposed observations from a single image are defined as consistent. Under this definition, selecting a single well-exposed image

for reconstructing the whole image would create a labeling with minimum energy. This selection is undesired since the information contained in other consistent images is left out of the average, thus degrading the SNR of the resulting irradiance estimates (see Fig. 3.9, top row). Instead, whenever two distinct sets are consistent, the set that produces lower-variance estimates is preferred, regardless of the set size. To encode such a preference, a variance potential $V(S_l)$ is introduced for assigning higher costs to groups that provide higher-variance estimates. The relative variance of each estimate is given by

$$V(S_l) = \sqrt{\frac{\sigma^2_{S_l}}{\sum_{S_m \in S(s)} \sigma^2_{S_m}}}, \tag{3.10}$$

where the variance of each group is approximated as $\sigma^2_{S_l} = (\sum_{i \in S_l} 1/t_i^2)^{-1}$.

3.7.1 Handling of Under- and Over-exposed Pixels

Note that no provisions are introduced in Eq. 3.9 for handling subsets of images that contain ill-exposed pixels, i.e., pixels that are over- or under-exposed, with the exception of subsets containing a single image.

However, when a color $v_i^k(p)$ is saturated, the corresponding $X_i^k(p)$ and $\sigma_{X_i^k(p)}$ are under-estimated in Eq. 3.1 and Eq. 3.4, respectively. Therefore, it is very unlikely that these saturated observations are consistent with other well-exposed observations. One can expect that our algorithm marks as inconsistent those subsets that contain both saturated and well-exposed observations. One can also expect that it marks as inconsistent those subsets containing only saturated observations from images with different exposure time. This is likely to occur since the irradiance estimates from saturated observations differ when the exposure time is different. For these reasons, no additional terms are included for penalizing subsets with saturated observations in our energy functional.

On the other hand, when observations are under-exposed, the light intensity is too low to be measurable above the camera's readout noise. In this case, the corresponding irradiance and variance estimates are not under-estimated but simply uncorrelated with the signal. In this case, the algorithm is likely to mark under-exposed observations as inconsistent with other well-exposed observations, and with under-exposed observations on images with different exposure time. Therefore, no additional terms are included for penalizing subsets with under-exposed observations.

Lastly, the observed colors $v_i^k(p)$ can fall below the camera's black level due to the effect of readout noise. This can lead to negative irradiance estimates $X_i(p)$ in Eq. 3.1. For handling this case, these negative values are still used for scoring the consistency in Eq. 3.6, but negative values of $X_i(p)$ are clamped to zero when computing the variance in Eq. 3.4, in a way that the variance estimates remain well defined.

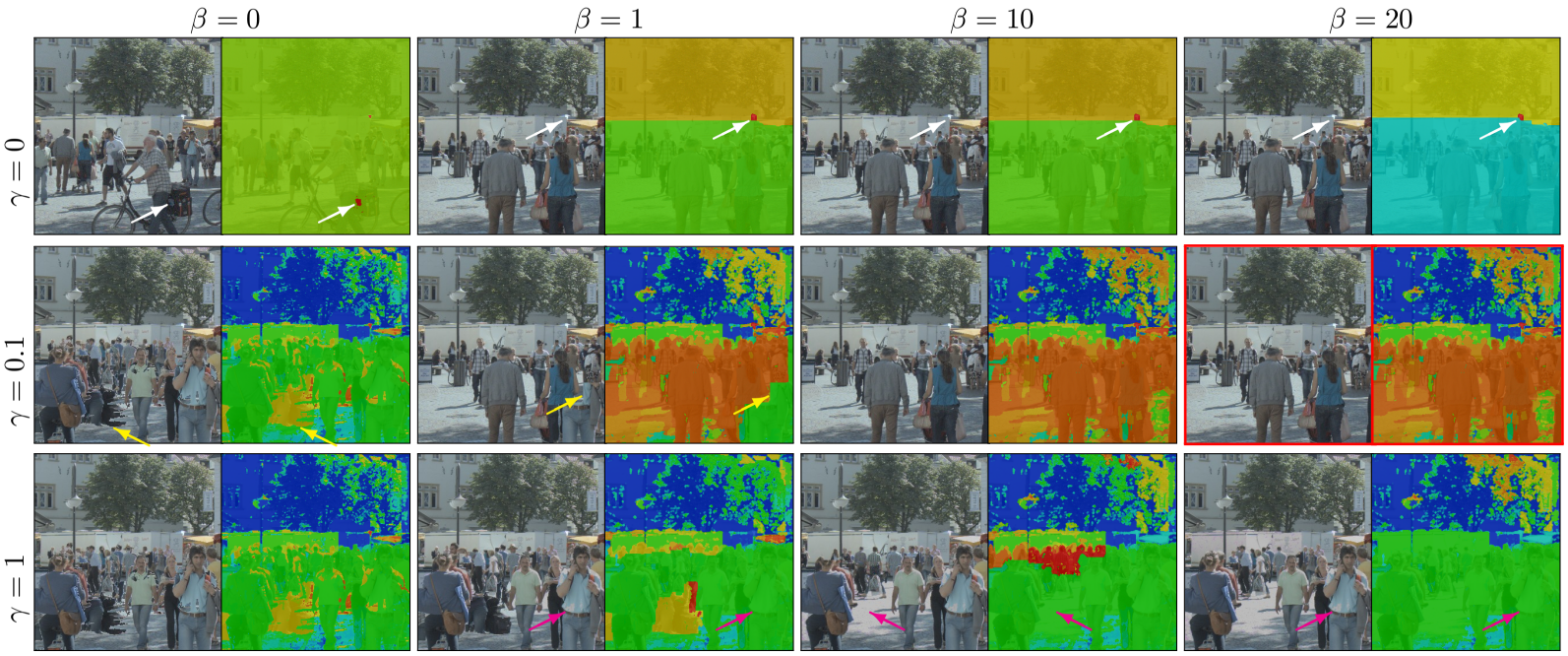


Figure 3.9: Effect of varying the parameters β and γ in Eq. 3.9. The parameters $\beta = 20$, $\gamma = 0.1$ are selected (enclosed in red) since they produce a good trade-off between low noise and spatial consistency. These parameters are kept fixed in all our experiments.

3.7.2 Parameter Selection

There are three hyper-parameters to be tuned in Eq. 3.9: The weight γ for the variance potential, the confidence value α of the consistency tests, and the weight β of the prior potential. The parameter γ was set at 0.1 in order to ensure that the variance potential in Eq. 3.9 produces an order-of-magnitude lower cost than the consistency potential. Accordingly, this design instructs the algorithm to prefer consistent subsets, but when presented with several consistent options, it should prefer the one with the least noise. The other two parameters were determined based on a performance evaluation using the *busy square* sequence (Fig. 3.10). The confidence α was set at 0.99, which provides a good trade-off between sensitivity and specificity of the ghosting detection, when compared to a manual annotation of the scene (see Sec. 3.8.5 for details). The parameter β was chosen to be 20, which is the lowest value that did not introduce visual discontinuities on the test sequence (see Fig. 3.9). Once determined, the parameters α, β, γ were fixed for all the experiments presented in this chapter.

Figure 3.9 shows the effects of varying parameters β and γ. In our preliminary experiments, variations of α did not affect the results significantly. When noisy subsets are not penalized ($\gamma = 0$; top row), the algorithm mostly selects a single image as source except for ill-exposed regions (pointed by white arrows), as only such regions are considered inconsistent. This behavior holds regardless of the weight β given to the prior potential. If noisy subsets are penalized mildly, i.e., less than inconsistent subsets ($\gamma = 0.1$; middle row), the remaining subsets of larger SNR (shaded in blue and green colors) are preferred provided that they are consistent, resulting in labelings that adapt more to the scene. In this configuration, as the weight β of the prior potential increases, visual discontinuities (marked by yellow arrows) are eliminated from the deghosted image (e.g. in $\beta = 10, 20$). When noisy subsets are penalized as much as inconsistent ones ($\gamma \geq 1$; bottom row), it becomes affordable to include objects that are partially ill-exposed (pointed by purple arrows) if they appear on the longest (less noisy) image. These results support our choice of γ.

3.7.3 Optimization and Final Reconstruction

To obtain a labeling F^* of minimum cost, we apply the expansion-move algorithm [Boykov01, Boykov04]. Using the resulting labeling, the final irradiance map is estimated as the weighted average

$$\hat{\mu}^k_{X(p)} = \frac{\sum_{i \in S(s)_{F^*(p)}} O_i(p) W_i(p) X^k_i(p)}{\sum_{i \in S(s)_{F^*(p)}} O_i(p) W_i(p)}, \tag{3.11}$$

where the weighting function $W_i = (\sum_{k \in \{R,G,B\}} \sigma^2_{X^k_i(p)})^{-1}$ is used in order to produce a result close to the maximum likelihood solution [Robertson03],

Camera	Sequence	Challenges	Gain
Canon PowerShot S5	Busy square (Fig. 3.10)	* Scene clutter * Large displacements	0.2417
	Flower shop (Fig. 3.12)	* Scene clutter * Large displacements	0.2390
	Food market (Fig. 3.13)	* Localized motion	0.2315
	Playground (Fig. 3.14)	* Localized motion	0.5978
	Traffic light (Fig. 3.15)	* Large dynamic objects * Large displacements	0.2343
	Café terrace (Fig. 3.20)	* Localized motion	0.2250
	Square at night (Fig. 3.21)	* High noise * Low light	0.4125
Canon EOS 550D	Christmas market (Fig. 3.17)	* Localized motion * Low light	1.67

Table 3.2: Summary of the cameras and test sequences used for experimental validation. The reported gain factors were estimated using the image-based method. The ground truth gain factor for the Canon S5 and 550D were g = 0.2394 and g = 1.87, respectively.

while at the same time applying identical weights to every color channel.

3.8 Experimental Validation

This section provides an experimental validation of the proposed HDR deghosting method on several real-world sequences. See Table 3.2 for a summary of the sequences. In addition, the ghost-detection accuracy of the proposed method is compared with the top performing state-of-the-art deghosting methods.

3.8.1 Experiment Setup

For the experiments, a compact digital camera (Canon S5IS, 10bit ADC) was used with gain factor set on ISO100. Results with an additional camera are provided in the following sections. The camera's black level ($L_0 = 32$) and readout variance ($\sigma_R = 2.655$) were estimated from an additional black frame following the method described in Sec. 2.2.4. The gain factor (Table 5.1) was estimated independently for every sequence using image-based calibration (see Sec. 3.4). Note that the gain needs to be estimated only once for any given camera model; the calibration was performed for each sequence independently in order to validate the robustness of the image-based calibration and deghosting method. For reference, the ground truth

gain factor (g = 0.2394) was also estimated using 36 flat-field images following the procedure described in 2.2.4. The flat-field images were obtained by placing the camera lens in front of a LCD display with the focus set to infinite distance.

The test scenes comprise eight different outdoor scenes that contained moving people or cars (see Table 3.2). The scenes were acquired during day and night time using a tripod. The camera was programmed using the CHDK toolkit [Doe12] to capture sets of five images at $\{0, -1, +1, -2, +2\}$ stops. The exposure time and aperture of the first image were determined automatically by the camera's light meter. The aperture was left constant on all images.

The camera was configured to save *raw* images (no demosaicing, white balancing, intensity mapping, or compression). Color images were generated by taking a green, red, and blue sample from each 4×4 block of pixels in the un-demosaiced raw image to set the color of a single pixel in the final color image. This leads to images of half the spatial resolution but with undistorted noise properties. The problem of performing demosaicing while keeping track of the effects on the image variance is left for future work.

After performing de-ghosting on the input images, white balancing was applied to the resulting HDR image using the factors specified by the camera. In order to show the results in this chapter, the resulting HDR images were tone mapped using the operators of Drago et al. [Drago03] (busy square, flower shop, food market, playground, café terrace), and Reinhard and Devlin [Reinhard05a] (traffic light, Christmas market, square at night).

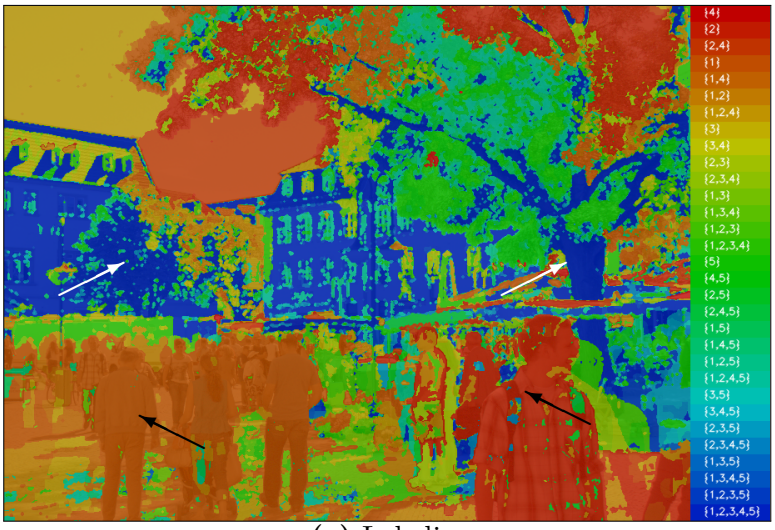
3.8.2 Results

The results presented in this section were computed with prior weight set to $\beta = 20$, Potts energy set to $\gamma = 0.1$, and confidence value $\alpha = 0.99$.

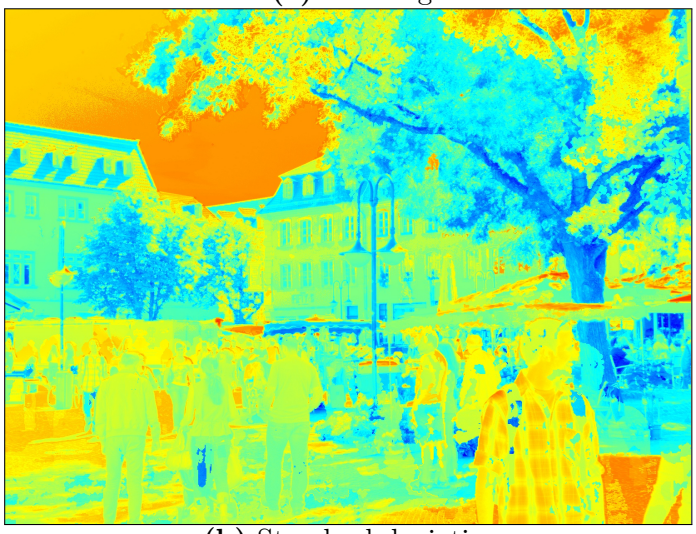
The sequences *busy square* (Fig. 3.10), *flower shop* (Fig. 3.12), *food market* (Fig. 3.13), and *traffic light* (Fig. 3.15) show how strong scene clutter can cause severe ghosting artifacts in a baseline HDR reconstruction, which includes every image into the irradiance average. In all these cases, the proposed algorithm successfully removed any ghosting artifacts from the final HDR image. In addition, the *square at night* sequence (Fig. 3.21) shows that our algorithm is also robust to high image noise occurring in low light conditions.

In the other hand, the sequences *playground* (Fig. 3.14), *Christmas market* (Fig. 3.17), and *café terrace* (Fig. 3.20) contain relatively small object displacements. Scenes showing this type of displacements are the target of the reference-based methods (Sec. 3.2.2) evaluated in Sec. 3.8.4. Even in sequences that favor reference-based methods, the proposed method produces better results, demonstrating its applicability to a wide variety of scenes.

Figure 3.10: De-ghosting of the *busy square* sequence: (top) Five input images with different exposure time; (middle) standard HDR reconstruction where inconsistent pixels are included in the average; (c) de-ghosted HDR image using the proposed method. The labeling is corresponding to this result is shown in 3.11.



(a) Labeling



(b) Standard deviation

Figure 3.11: Visualization of the HDR deghosting process for the *busy square* sequence (Fig. 3.10): **(a)** Labeling corresponding to the subset per pixel selected by the algorithm (out of 31 possible subsets); the subset labels on the right hand side are sorted from top to bottom in order of decreasing variance of the irradiance estimates. In general, the algorithm selects several images for static image regions whose colors are consistent (pointed by white arrows), whereas it selects single images for dynamic image regions where it is likely that no image is consistent to any other (pointed by black arrows). **(b)** Standard deviation of the deghosted HDR image, where larger subsets generally produce irradiance estimates with lower noise; blue denotes low noise, red high noise.

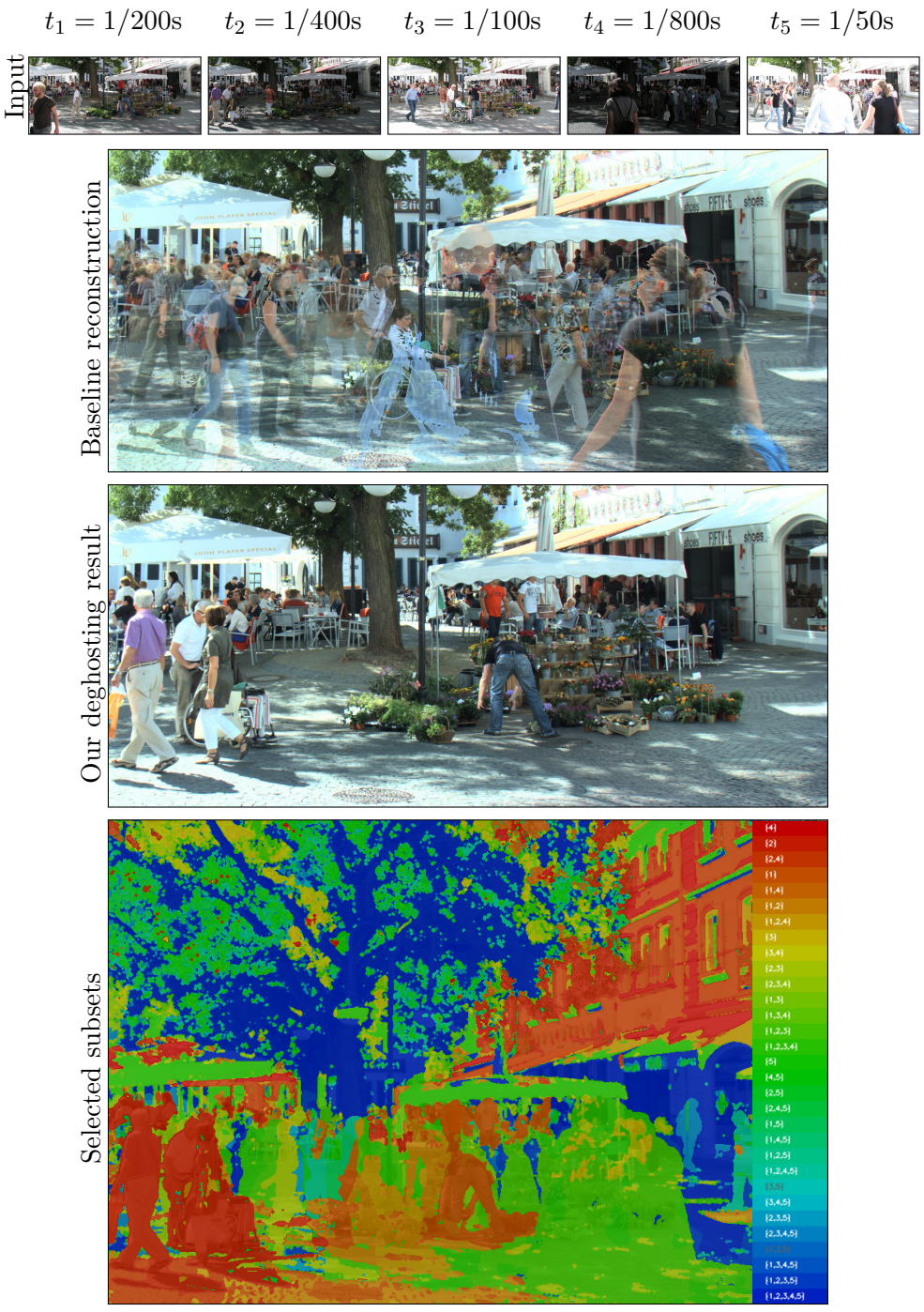


Figure 3.12: De-ghosting of the *flower shop* scene: (top) Input images; (second row) Naïve averaging of the input images which produces severe ghosting artifacts; (third row) ghost-free result of our method; (bottom) image subsets selected by the deghosting algorithm.

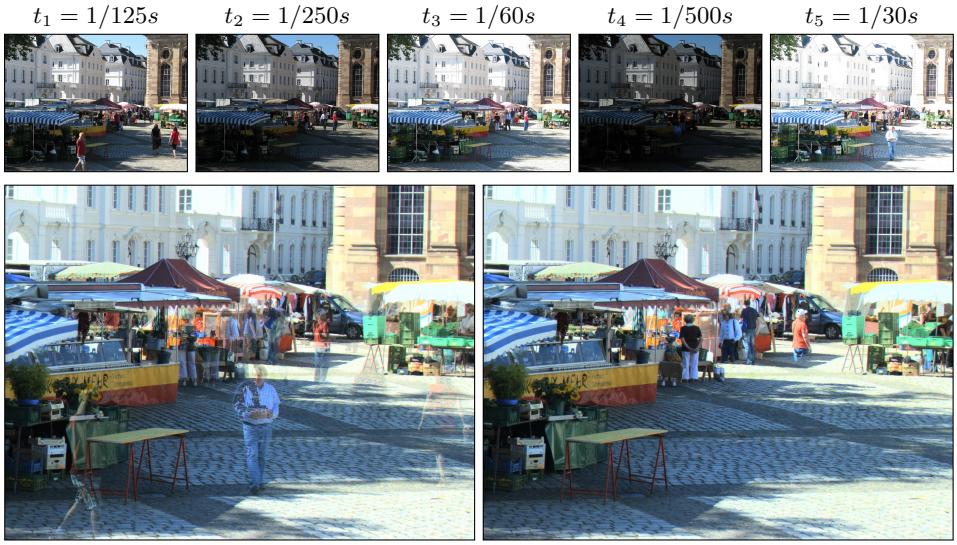


Figure 3.13: De-ghosting of the *food market* scene: (top) Five input exposures of a dynamic scene showing a pedestrian zone; (bottom-left) Standard HDR reconstruction (no deghosting); (bottom-right) Ghost-free HDR reconstruction (tone-mapped) obtained by the proposed algorithm.

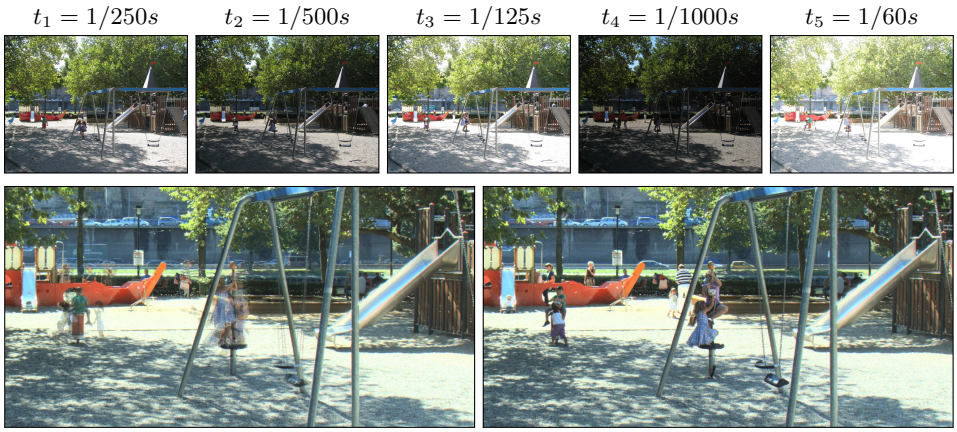


Figure 3.14: De-ghosting of the *playground* scene: (top) Five images of a scene that shows people undergoing localized motion (i.e., no large displacements); (bottom-left) Standard HDR reconstruction exhibiting several ghosting artifacts; (bottom-right) Ghost-free result of the proposed algorithm.

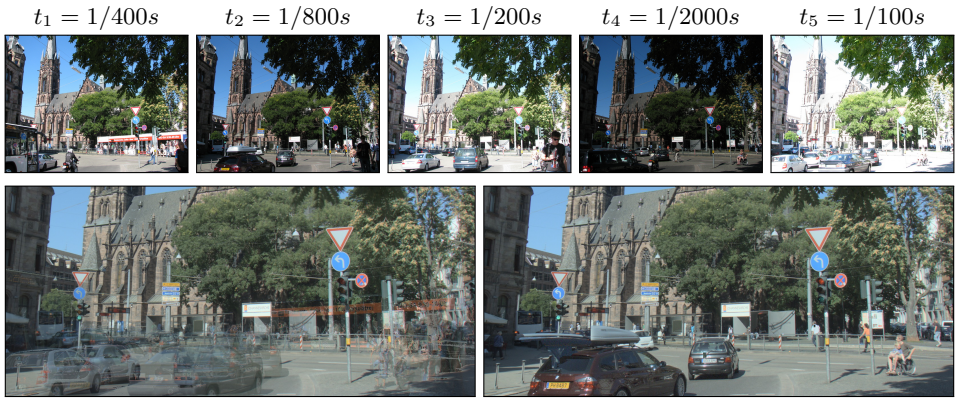
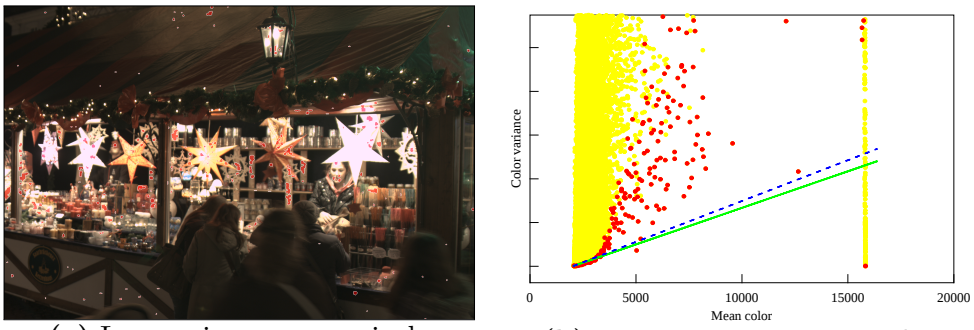


Figure 3.15: De-ghosting of the *traffic light* scene: (top) Five exposures of a cluttered scene showing both small and large objects undergoing large displacements; (bottom-left) Standard HDR reconstruction displaying several ghosting artifacts; (bottom-right) Tone-mapping of the ghost-free HDR image obtained by our algorithm.

In abstract, the proposed method prevents ghosting by selecting a consistent set of images at each pixel while at the same time encourages the selection of seamless transitions between adjacent pixels assigned to different image sets. Although the final labeling contains several of such transitions, in most cases the final result does not display any boundary artifacts (the situations where this might occur are discussed below). These transitions are illustrated in Fig. 3.11a where an example of the labelings produced by our method is shown. This example shows how the algorithm can correctly select single images for estimating the irradiance of pixels that display a different object in every image (e.g. for the pedestrian zone, see black arrows), and also select larger sets of images for the estimation at pixels that show static objects (e.g. on the facades, see white arrows). Other examples of the labelings computed by our method are shown in Fig. 3.12 and Fig. 3.17.

Additionally to producing ghost-free HDR images, the noise on each HDR image can be also predicted. This is possible since the camera noise parameters are calibrated prior to the reconstruction. Using this noise prediction, any HDR reconstruction procedure (e.g. the procedure presented in Chapter 4) can also provide estimates of the noise distribution of the final HDR image (see Fig. 3.11b for an example). These estimates can be used as input for other tasks such as HDR image denoising (see 4.6.2 on the next chapter).



(a) Low-variance super pixels **(b)** Mean-variance scatter-plot

Figure 3.16: Image-based calibration of the *Christmas market* sequence. This sequence was acquired using a different test camera than the rest of the results presented in this chapter. Despite the scarcity of low-variance super pixels in this low light scene, the camera gain was accurately calibrated when compared with the ground-truth.

Experiments with Additional Cameras

Figures 3.16 and 3.17 show the deghosting result of an additional sequence named *Christmas market*. This sequence contains three exposures of a scene that shows dynamic objects undergoing local motion. This sequence is acquired using a Canon EOS 550D set at ISO400; this camera is *different* from the one used for the remaining experiments in this chapter. Regardless of the low light in the scene, the gain estimate obtained with our image-based calibration ($g = 1.67$) is very close to the one obtained using flat-field calibration ($g = 1.87$). The successful calibration and the corresponding final deghosted image (Fig. 3.17) demonstrate that the proposed gain calibration method generalizes well to different cameras.

3.8.3 Refinement of Potential Semantic Inconsistencies

In some cases, it is possible that the proposed algorithm produces results that are plausible color-wise but semantically incorrect. For instance, the person with a checkered shirt appears twice in the HDR image (Fig. 3.10–bottom), first on the left, then on a close up on the right, as he was photographed at different times in the fist two input images.

A different type of semantic artifacts is illustrated in Fig. 3.18, where the HDR image is reconstructed using only the first three exposures of *busy square*. In the result, the rightmost person is only half included. This occurred as the data term penalized the dark under-exposed pants and the bright over-exposed ground behind, causing the lower part of the person's body to be excluded during the optimization.

Figure 3.17: Deghosting result on the *Christmas market* sequence. The result our algorithm (third row) effectively avoids ghosting artifacts (second row) even though the motion of the objects in the scene is highly localized. The selected subsets (bottom row) show that several images are used for reconstructing the static parts of the scene (blue, green) but only individual images are used for reconstructing the dynamic objects (red, yellow).

Figure 3.18: De-ghosting of the *busy square* using only the first three images shown in Fig. 3.10–top: (left) standard HDR reconstruction; (right) de-ghosting result. The legs of the rightmost person are not selected by the algorithm as the pants are under-exposed, and the background behind them is over-exposed. This type of semantic mistakes can be corrected interactively by the user (Fig. 3.19).

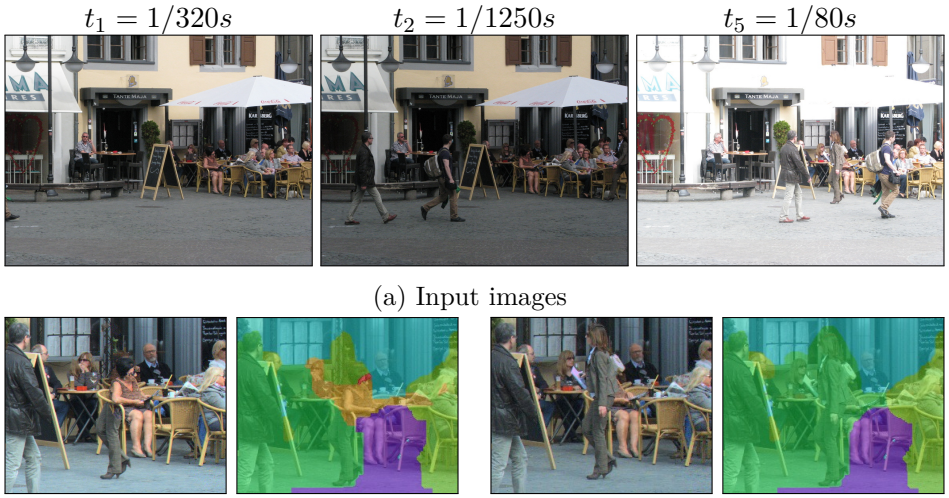
As in the last example, it is possible that all objects in a given image location are partially ill-exposed. I claim that these type of artifacts cannot be corrected automatically since the choice of including ill-exposed regions for the sake of consistency cannot be done without knowing the exact image extent of each object. For this reason, a user interface was constructed where the user can correct such semantically inconsistent images. This solution is exemplified in Fig. 3.19. In this case, the legs but not the torso of a person was included in the final HDR image, since the upper part contained over-exposed pixels (Fig. 3.19b). Using a simple interface, the user can correct this inconsistency by painting the correct labels over the automatically generated labeling, in a way that the resulting HDR image becomes semantically correct (Fig. 3.19c).

An automatic alternative to prevent semantic mistakes in the resulting HDR images is to perform scene-specific object detection (e.g. of people or cars). The image span of the detected objects can be used to constraint the energy function such that no transition between image subsets occurs at the interior of these objects. The analysis of this alternative is left for future work.

3.8.4 Comparison with Reference-based Methods

This section compares the proposed deghosting method with the state-of-the-art methods of Sen et al. [Sen12], and Zimmer et al. [Zimmer11] on the *busy square* sequence using their own implementations. Each of these methods specify one of the input images as reference, whose dynamic range is enhanced using the corresponding content on other input images.

The method of Sen et al. finds correspondences between the reference and the remaining input images using PatchMatch [Barnes09]. This search may be negatively impacted by the fact that the reference image has a



(a) Input images

(b) Inset of the automatic result

(c) Result after editing

Figure 3.19: De-ghosting of a scene where semantic artifacts arise. In this result, the legs of a pedestrian are selected, but her upper body is not, as she carries a piece of paper that is over-exposed (a). Note that the algorithm selects a transition that is color consistent. Such semantic inconsistencies are corrected using an user interface where the user can edit the labeling generated automatically (b).

low dynamic range, thus regions that are ill-exposed or contain high noise might not be properly matched to other exposures. This is demonstrated in Fig. 3.20, where the dynamic range of over-exposed regions could not be enhanced (indicated by arrows). Additionally, Fig. 3.21 shows that strong noise in the reference may preclude finding corresponding regions in other images leading to a very noisy HDR image. In contrast, our method is designed to select sets of images that are both consistent and have low noise, resulting into HDR of improved quality and less noise.

In the other hand, Zimmer et al. establish correspondences using optical flow which fails on objects in our data sets that undergo large displacements, or non-linear local displacements. This failure case is shown on the person in Fig. 3.22, where ghosting artifacts are introduced after two instances of a person undergoing local motion cannot be properly aligned. In contrast, our method selects a single (self-consistent) image in this case, thus preventing the introduction of ghosting artifacts.

3.8.5 Comparison with Ghost-detection Methods

This section compares performance of the proposed ghost-detection method against the best performing methods reported by Sidibé et al. [Srikantha12]. The top four detection methods according to their sensitivity score are:

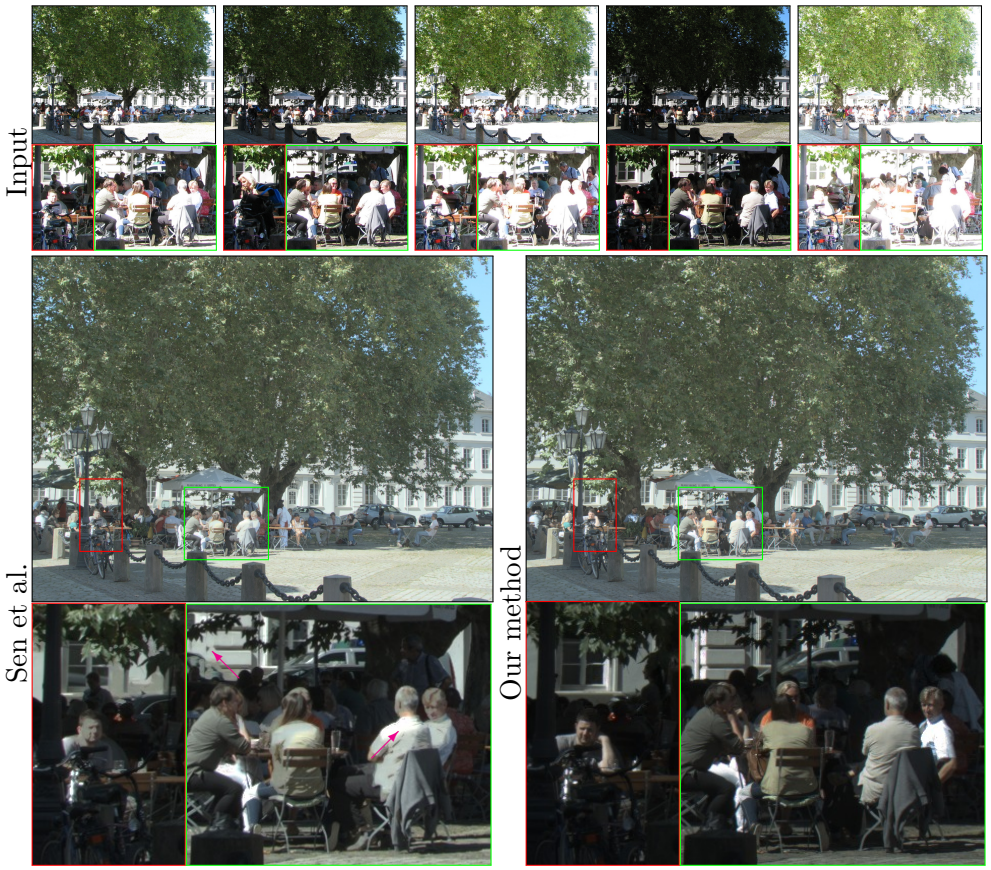


Figure 3.20: Comparison to Sen et al. on the *Café terrace* sequence (top). The first image was selected as reference by the method of Sen et al. Here, their method encounters difficulties extending the dynamic range of ill-exposed regions, which results in a washed-out appearance (bottom-left, indicated by arrows). In contrast, our method automatically selects well-exposed sources for every region.

Figure 3.21: Comparison with the method of Sen et al. on the *Square at night* sequence (top). The second exposure was selected as reference for Sen et al.'s method. Due to noise, their method finds few similar patches in other exposures. This implies that the dynamic range cannot be effectively extended using other input images (bottom-left). Our method selects consistent sources with as low variance as possible, preventing the appearance of noise in the result (bottom-right).

(a) **(b)** **(c)** **(d)**

Figure 3.22: Comparison with the method of Zimmer et al. on the *busy square* sequence: **(a)** Reference image, **(b)** optical-flow alignment of an additional input image to the reference, **(c)** result after HDR reconstruction using (a) and (b), and **(d)** our result.

Grosch [Grosch06], Sidibé et al. [Sidibé09], Heo et al. [Heo10], and Pece and Kautz [Pece10]. These de-ghosting methods work in two stages: Detecting the moving objects in the images, and reconstructing the final image using only the static parts. Since the resulting motion detection is often noisy, these methods use different regularization techniques to improve the corresponding motion masks (e.g. Gaussian smoothing, morphological operations, or MRF priors). These regularization techniques depend on the priors assumed over the properties of the moving objects. However, any given regularization technique could be applied to any of the detection methods. Therefore, in order to exclude the effect of different regularization strategies (i.e., of different priors), the comparison is performed only on the motion detection part of every method (see Fig. 3.23 and Table 3.3).

For the comparison, a manual segmentation was performed of the moving objects in the first two images of *busy square* (Fig. 3.23a, 3.23b). The resulting mask is used as reference for scoring the result of each ghost-detection method (Fig. 3.23c).

Table 3.3 summarizes the sensitivity and specificity of the consistency mask produced by each method. Among previous methods, the best sensitivity (43.6%) was achieved by the method of Grosch (Fig. 3.23e), which thresholds the absolute irradiance difference between the images. As the author does not report any specific threshold, it was set to the median plus three times the median average difference of the two images, which was experimentally found to give the best results. This method also achieves the lowest specificity (93%). The resulting mask fails to detect motion in regions with lower irradiance, and detects false motion in bright regions such as the sky.

Detection strategy	Sensitivity	Specificity
Proposed method, $\alpha = 98.0\%$	**0.513784**	0.957435
Proposed method, $\alpha = 99.0\%$	0.500532	0.981069
Proposed method, $\alpha = 99.9\%$	0.467973	0.994732
Absolute difference [Grosch06]	0.43628	0.92950
IMF probability [Heo10]	0.25692	0.93983
Monotonic ordering [Sidibé09]	0.24698	0.99441
Median threshold [Pece10]	0.15810	**0.99998**

Table 3.3: Comparison of the accuracy of the ghost-detection methods.

The method by Heo et al. achieves lower sensitivity (25.7%) but higher specificity (94%) than Grosch's (Fig. 3.23f). Their method is based on detecting regions where the intensity mapping function (IMF) has a low probability. The method fails to detect motion in this scene due to the high proportion of moving objects, which makes it difficult to properly estimate the IMF probability.

The method of Sidibé et al. (Fig. 3.23g) and Pece and Kautz (Fig. 3.23h) achieve the highest specificity (99.4% and 99.9%, respectively) but the lowest sensitivity (24.7% and 15.8% respectively). This can be explained as both methods are based on invariants (monotonic increase of output value with the exposure time, and ordering with respect to the median irradiance value of the scene, respectively) that are satisfied whenever two pixels correspond to the same light intensity. However, these invariants are not always violated by moving objects, which leads to a lower sensitivity.

The proposed method was tested with confidence values $\alpha = \{0.98, 0.99, 0.999\}$. In all cases, the sensitivity of our detector (51.3%, 50%, 46.8%, respectively) was the best among all methods, while its specificity (95.7%, 98.1%, 99.5%, respectively) was only lower than the methods based on invariants, which all have very low sensitivity. Our method achieved a good trade-off between sensitivity and specificity at $\alpha = 0.99$ with a sensitivity and specificity of 50% and 98%, respectively (Fig. 3.23d). The low nominal value sensitivity (e.g. 50%) is explained by the fact that the ground-truth segmentation was constructed manually with the objective of reconstructing ghost-free HDR images. For this reason, the moving regions were masked in a conservative way (i.e., over-segmentation was preferred over under-segmentation), and therefore, they might include static parts of the scene. Nevertheless, when compared with previous methods, the sensitivity of our method is always higher, and the resulting mask represents better the moving objects in the scene (see Fig. 3.23d).

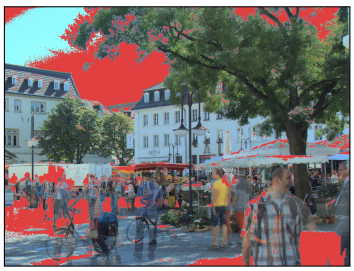
(a) First image, $t_1 = 1/500s$

(b) Second image, $t_2 = 1/1000s$

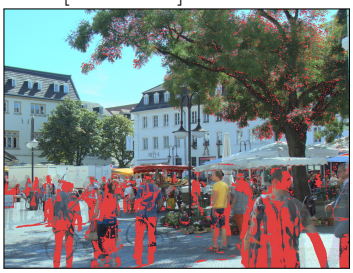
(c) Ground truth (manual)

(d) Our method, $\alpha = 99\%$

(e) Absolute difference [Grosch06]



(f) IMF probability [Heo10]



(g) Monotonic ordering [Sidibé09]

(h) Median threshold [Pece10]

Figure 3.23: Comparison between our method and of previous methods: Our method (d) performs a more accurate detection of the moving objects between two images of a scene (a–b), when compared with a manual ground-truth segmentation (c). The accuracy of state-of-the-art ghosting detection methods (e–h) is always lower than ours. The numerical accuracy of each method is reported in Table 3.3.

3.9 Discussion

This section discusses the advantages and limitations of the proposed method for ghosting detection and removal.

Cluttered Scenes

As illustrated in Sec. 3.8, the proposed algorithm can produce plausible de-ghosting results for image sequences of scenes with severe clutter. To the best of my knowledge, this is the first method in the literature that succeeded under such level of clutter; all previously published methods use test sequences with only few moving objects over a clean background. This level of performance is achieved by defining robust tests for motion detection that are evaluated within a global energy minimization framework.

Length of the Exposure Sequences

Our method performs a robust motion-detection test over every possible subset of images (i.e., a factorial number of subsets) to verify their consistency. This implies that the proposed method is restricted to short exposure sequences. However, image sets of three to five exposures can still be de-ghosted within minutes. In practice this does not pose a limitation for the application of our method as exposure sequences are often recorded using a technique called *exposure bracketing*, where the camera automatically captures three photographs of the scene, one with suitable exposure settings, plus two additional under-exposed and over-exposed images. Therefore, in this practical scenario of exposure bracketing, our method is the best available HDR reconstruction method.

Alternatives for extending our method to be applicable to longer exposure sequences include setting a reference low-dynamic-range image so only comparisons with the reference image are required, and using faster but less accurate optimization procedures. However, these strategies might eventually compromise the quality of the de-ghosting results.

Extreme Exposure Differences

Our motion detection method performed very well even under extreme exposure differences: The exposure sequences had up to four stops between the longest and shortest exposure; for the tested camera, this is the maximum possible exposure difference that allows the same pixel to be well exposed on both images. This detection performance suggests that the noise model correctly accounts for the variance differences between images, such that the measured values on each of the images become comparable.

Assumptions about the Input Sequences

The proposed method makes two assumptions about the input. First, it assumes that the camera position and the configuration of the optics are constant across exposures. This assumption warranties that each pixel measures the same incoming irradiance on every image. Second, the method assumes that the variance of a given color measurement can be predicted. In practice, this can be done through an accurate noise model for CCD/ CMOS sensors whenever raw sensor output is available. Such a model can be easily calibrated for every camera and a method for performing such calibration using the input images is presented.

Potential Semantic Inconsistencies

The results presented in Sec. 3.8 can contain semantic inconsistencies, such as half-included objects. These inconsistencies occur as the algorithm does not require information regarding the boundary of the objects in the scene that could be used to reliably avoid their partial inclusion in the result. In scenarios where object boundaries are known (or can be reliably estimated), these artifacts can be easily penalized in the energy function.

In general, this type of semantic inconsistencies can occur in two situations: If the color at the boundary between image subsets is similar enough so their difference can be explained by noise, or if the omission of the boundary leads to the inclusion ill-exposed regions. The first situation arises from an intrinsic limitation of the chosen method: Measurements can only be compared up the noise level of the signal. This limitation could be addressed, for instance, by encouraging boundaries to occur at edges of the image. However, these strategies require non-metric priors that cannot be properly optimized using the graph-cuts framework.

The second situation arises when all the objects occupying a given image region on different exposures are ill-exposed. In this case, no object can be fully included without introducing ill-exposed pixels in the result. Resolving this situation requires deciding whether it is preferable to include ill-exposed regions or partially visible objects. In general, this decision depends on the application, and it should be taken by the user, for instance, through the proposed user interface.

3.10 Conclusion

This chapter presented a new method for reconstructing HDR images of highly dynamic scenes, i.e. scenes that contain a large number of moving objects. The proposed method takes advantage of a previously unexploited strategy: The capability of predicting the noise level of any of the input images. Such prediction is performed based on an accurate camera noise model

that approximates well the noise behavior of CCD/CMOS digital sensors. For calibrating the noise model, a new simple method was proposed for estimating the camera gain factor from the input images, therefore, enabling the automatic prediction of the image noise range. The resulting prediction can be used to accurately detect sets of images that are consistent, i.e. that display the same object on a given image region. In addition, the noise prediction allows selecting among different consistent sets those that have lower noise. Once such sets are detected, the resulting HDR image will not contain any ghosting artifacts caused by averaging images of different dynamic objects.

The selection of consistent image sets for every pixel is a combinatorial problem that is modeled using Bayesian inference with Markov priors and solved efficiently using graph cuts. The accuracy of the resulting method was experimentally found to be the best among the state-of-the-art methods. In addition, our method is the first to be shown to perform well in extremely challenging scenarios which have not been previously demonstrated in the literature, i.e. in scenes with low light, and in dynamic scenes that contain a large number of moving objects.

CHAPTER 4

Noise-optimal HDRI Reconstruction

4.1 Introduction

In Chapter 3, it was shown that the ability to predict the noise level in an image can be used to reconstruct HDR images of dynamic scenes without suffering from ghosting artifacts. In this chapter, I show that the camera noise model can be used to further improve the quality of the resulting HDR images. This is demonstrated in three ways: By defining a noise-optimal weighting function for combining the irradiance estimates provided by each image, by planning a exposure sequence that would produce an HDR image with a minimum desired signal-to-noise ratio, and by de-noising the final HDR image according to the predicted noise distribution of the image. These applications and the models that support them were published in [Granados10].

As discussed in Sec. 3.1, it is possible to recover the full dynamic range of a scene by averaging a set of images with taken different exposure. The resulting HDR image is obtained by transforming the input images from the digital domain to the irradiance domain (Eq. 3.1), and computing their average (Eq. 3.2). During the average, a weighting function is used to account for the potential differences in the accuracy of the irradiance measurements provided by each input image. However, the weighting functions proposed in the literature (discussed in Sec. 4.2) do not consider the individual noise sources involved in the camera acquisition process (except [Tsin01] and [Kirk06]), and therefore, they provide irradiance estimates with sub-optimal signal-to-noise ratio. This issue is especially relevant when the HDR images are

not required for visualization but as accurate measurements of the physical irradiance in the scene. To address this issue, in this chapter a weighting function is proposed that is optimal in the least-squares sense (Sec. 4.3). The optimality of the proposed weighting holds under the assumption that the camera noise follows a Gaussian distribution. Under this assumption, the optimal weighting function corresponds to the inverse of the variance, which can be predicted using a calibrated noise model. Unlike previous methods, the proposed weighting takes into account temporal noise sources and spatial noise sources (see Sec. 2.2). The resulting noise model can be used to characterize the scenarios where well-establish methods for HDR reconstruction [Mitsunaga99] perform unsatisfactorily (Sec. 4.4). The application of a more accurate noise models results in a higher reconstruction performance than previous methods. This demonstrated in Sec. 4.5 where it is empirically shown that the signal-to-noise of the irradiance reconstruction obtained by our method is the best among all currently available approaches.

The new optimal reconstruction method has several important practical implications. These are discussed in Sec. 4.6. First, the underlying noise model can also be used to help planning the acquisition process of HDR images. If the irradiance distribution of the scene is known (or assumed), the noise model can be used to define an exposure sequence that properly samples the irradiance range. For instance, the user can provide the desired minimum signal-to-noise of the HDR image, and an optimization algorithm can devise an exposure sequence that satisfies this constraint. This application is presented in Sec. 4.6.1. Second, the noise model can be used to further improve the quality of a given irradiance reconstruction. Since the variance of the irradiance is also estimated during the reconstruction process, it can be used to de-noise the final image in a noise-optimal way, i.e. without incurring in under- or over-smoothing. This application is presented in Sec. 4.6.2.

4.2 Previous Work

Several algorithms have been proposed for reconstructing an HDR image from a sequence of low dynamic range (LDR) images [Debevec97, Mann01, Mann95, Mitsunaga99, Reinhard05b, Robertson03, Tsin01]. These methods estimate an HDR image where the pixel values are proportional to the incident irradiance X. During the reconstruction, the inverse of the camera response function f^{-1} is also recovered. This function maps a digital output v to its inducing exposure $X\mathrm{t}$, where t is the exposure time of the image (see Eq. 3.3).

The final irradiance is estimated as a weighted average of the irradiance estimates provided by each image in the sequence (Eq. 3.2). However, several different weighting functions $W_i(p)$ have been proposed in the HDR recon-

Method	Type	Weighting
Mann & Picard [Mann95]	Quantization	$\frac{1}{\frac{d}{dv}(\log g(v))}$
Debevec & Malik [Debevec97]	Hat	$\min(v-v_{min}, v_{max}-v)$
Mitsunaga & Nayar [Mitsunaga99]	SNR	$\frac{g(v)}{g'(v)}$
Reinhard et al. [Reinhard05b]	SNR·Hat	$\frac{g(v)}{g'(v)}\left[1-\left(\frac{v}{v_{mid}}-1\right)^{12}\right]$
Robertson et al. [Robertson03]	Variance	$\frac{t^2}{\frac{d}{dv}(\log g(v))}$
Tsin et al. [Tsin01]	St. dev.	$\frac{t}{\hat{\sigma}_{g(v)}}$
Kirk & Andersen [Kirk06]	Variance	$\frac{t^2}{g'(v)^2\sigma_v^2}$

Table 4.1: Weighting functions for HDR reconstruction. Here, v is the camera digital output, t is the exposure time, and $g(v) \equiv \mathrm{f}^{-1}(v)$ is the inverse of the camera response function.

struction literature. The formulation of each weighting is listed in Table 4.1, and their shape is illustrated in Fig. 4.1. I will discuss them next.

In their seminal paper, Mann and Picard [Mann95] assign a weight to each digital output value according to the derivative of the inverse camera response. This design is motivated by the observation that the quantization error is lower for output values where the response function is steep, as a narrow range of irradiances is sampled using a larger number of digital values. The derivative of the camera response function is computed in a logarithmic scale in order to ensure that the resulting quantization error is perceptually uniform over the digital output range.

Debevec and Malik [Debevec97] propose a hat function that assigns higher weights to values in the middle of the digital output range, and lower weights to values in the extrema of the range. This design is motivated by the desire to avoid the inclusion of under- and over-exposed values in the irradiance average.

Mitsunaga and Nayar [Mitsunaga99] propose a weighting function that is designed to maximize signal-to-noise ratio of the resulting irradiance estimate. However, since they consider the camera noise behavior to be unknown, they assume that the variance of the digital output values is constant across the output range. In general, this assumption is invalid since the camera noise depends on the light intensity (see Sec. 2.2). Still, their weighting can behave optimally under specific conditions (Sec. 4.4). Reinhard et al. [Reinhard05b] extend the Mitsunaga-Nayar weighting by multiplying it by a hat function that reduces the importance of under- and over-exposed values.

Robertson et al. [Robertson03] assume a Gaussian distribution of the

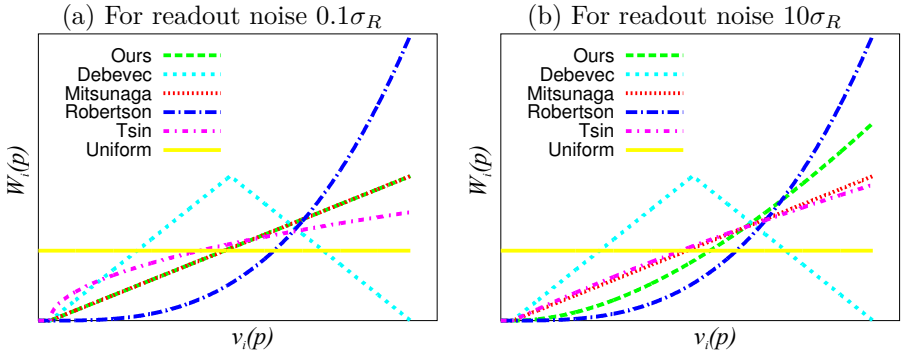


Figure 4.1: Shape of the existing weighting functions for irradiance reconstruction under the effect of two different readout noise levels (relative to a baseline level σ_R). The x-axis corresponds to the output values v_i induced by the same incident irradiance on images with increasing exposure time. The y-axis displays the weight assigned by the existing methods.

camera noise, and they are the first to propose a probabilistic approach to derive an optimal weighting function. Their derivation requires having an estimate of the variance of the digital output values. They approximate this variance using the Mann-Picard weighting, thus accounting only for quantization noise. The resulting weighting decreases linearly with the digital output variance and increases quadratically with the exposure time. They also smooth the weighting function towards zero at the extrema of the output range in order to exclude under- and over-exposed pixels from the average.

Tsin et al. [Tsin01] are the first to use a calibrated camera noise model in the problem of HDR reconstruction. They propose a weighting function that is inversely proportional to the standard deviation of the output value, which is predicted by the noise model. Since the irradiance is constant for a given pixel on all exposures times, the resulting weighting is equivalent to the signal-to-noise ratio of the irradiance estimate. However, the inverse of the standard deviation is still a sub-optimal weighting under the assumption of Gaussian noise; the inverse of the variance is the optimal weighting (Sec. 4.3).

Similarly to Robertson et al., Kirk and Andersen [Kirk06] follow a probabilistic approach to derive the optimal weighting function, they use a calibrated camera noise model to predict the variance of the digital output. However, their variance estimates are derived directly from the camera output, which transfers the measurement uncertainty into the weighting function. This condition is shared by all previous methods. In addition, Kirk and Andersen apply a noise model that does not account for spatial sources.

In the next section, I present an extension of the Kirk-Andersen weighting. The proposed weighting adopts a more rigorous camera noise model

that also accounts for spatial noise, such as dark current and photo-response non-uniformity, which dominate the noise in long exposures and in values close to saturation, respectively [Janesick85]. In addition, the proposed weighting function only depends on the current irradiance estimate as opposed to depending on the digital output values in each image. This is done with the objective that the noise differences between images do not affect the weighting function. This choice leads to an iterative method for estimating the irradiance and its variance. The resulting noise-optimal weighting function has been found to have the best performance among all existing methods (Sec. 4.5). This result has been confirmed in preliminary third-party evaluations [Aguerrebere12].

4.3 Optimal Weighting Function

In this application of the camera model, the goal is to obtain the best possible irradiance estimate, i.e., that of minimum variance, from a set of n measurements $\{(v_i(p), b_i(p), \mathrm{t}_i)\}_{i=1\ldots n}$, where $v_i(p)$ is the image color at pixel $p \in \mathcal{I}$, $b_i(p)$ is color in the dark frame, and t_i is the exposure time. Only the exposure time is allowed to vary between measurements, whereas all other camera settings (ISO value, aperture size, focal length) are left fixed. In order to analytically solve for a minimum variance irradiance estimate, the measurements are assumed to come from *raw* images, i.e., images obtained before any in-camera processing (e.g. dark frame subtraction, demosaicing, denoising, white balancing, and compression).

In the following, I assume that $X_i(p)$ has a Gaussian distribution with mean $\mu_{X(p)}$ (equal for all exposures) and variance $\sigma^2_{X_i(p)}$ (different for every exposure). The suitability of this assumption is discussed in Sec. 4.5.3. Due to saturation, which occurs when the sensor capacitor cannot accumulate more charge, Eq. 2.23 is only valid for values $v < \mathrm{L}_{sat}$, where L_{sat} is the saturation limit (defined in Sec. 2.2.4). Naturally, noise introduces uncertainty on the classification saturated values. Therefore, it is necessary to define a probability mass $O_i(p)$ that an observed pixel in image i is not saturated.

Let $\Pr(X_i(p)|\mu_{X(p)}, \sigma^2_{X_i(p)})$ be the conditional probability density of an observation $X_i(p)$. This function can be described as a *blending* of the unclipped probability density (under no saturation) and a uniform probability, where the blending factor is given by the saturation probability:

$$\begin{aligned}\Pr(X_i(p)|\mu_{X(p)}, \sigma^2_{X_i(p)}) = (1 - O_i(p)) \Pr_{uniform} + \\ O_i(p) \cdot \Pr_{unclipped}(X_i(p)|\mu_{X(p)}, \sigma^2_{X_i(p)}).\end{aligned} \tag{4.1}$$

Since the goal is to reconstruct the mean radiance $\mu_{X(p)}$ with the lowest variance from a set $\{X_i(p)\}_{i=1\ldots n}$ of independent measurements, the condi-

tional probability of X is computed as the joint probability

$$\Pr(X(p)|\mu_{X(p)}, \sigma^2_{X_1(p)}, \ldots, \sigma^2_{X_n(p)}) = \prod_{i=1}^{n} \Pr(X_i(p)|\mu_{X(p)}, \sigma^2_{X_i(p)}). \quad (4.2)$$

The maximum likelihood estimate for X is given by

$$\hat{\mu}_{X(p)} = \arg\max_{\mu_{X(p)}} \prod_{i=1}^{n} \Pr(X_i(p)|\mu_{X(p)}, \sigma^2_{X_i(p)}). \quad (4.3)$$

As this density function is typically convex, $\hat{\mu}_X$ can be iteratively approximated from n images by Newton estimation, starting from an initial averaged estimate $\hat{\mu}_{X(p)} = (\sum_i O_i(p)X_i(p))/(\sum_i O_i(p))^{-1}$.

In the other hand, an analytic solution of Eq. 4.3 can be derived by *ignoring* all pixels that are close or beyond the saturation limit by setting $O_i(p) = 1$ for $v < \mathrm{L}_{sat} - k\sigma_R$, and zero otherwise. The constant k is set to six standard deviations of the readout noise. Under this assumption, Eq. 4.3 simplifies to

$$\hat{\mu}_{X(p)} = \arg\max_{\mu_{X(p)}} \prod_{i \in S(p)} \Pr_{unclipped}(X_i(p)|\mu_{X(p)}, \sigma^2_{X_i(p)}), \quad (4.4)$$

where $S(p) \subseteq T$ is the set of images where the pixel p is well exposed. From Eqs. 4.4, 2.23, and 2.24 the maximum likelihood estimate is given by

$$\hat{\mu}_{X(p)} = \frac{\sum_{i \in S(p)} \frac{1}{\sigma^2_{X_i(p)}} X_i(p)}{\sum_{i \in S_j} \frac{1}{\sigma^2_{X_i(p)}}}, \quad (4.5)$$

with variance

$$\hat{\sigma}^2_{\mu_{X(p)}} = \frac{1}{\sum_{i \in S(p)} \frac{1}{\sigma^2_{X_i(p)}}}. \quad (4.6)$$

Analogously, from Eq. 2.25 and 2.26, estimates $\mu_{D(p)}$, $\sigma^2_{D_i(p)}$ can be obtained for the dark current.

From Eq. 3.2 and 4.5, it follows that the optimal weighting function for HDR reconstruction is

$$W_i^{(\mathrm{opt})}(p) = \frac{1}{\sigma^2_{X_i(p)}}. \quad (4.7)$$

After plugging in Eqs. 2.20, 2.22, and 2.24, the variance estimate is given by

$$\sigma^2_{X_i(p)} = \frac{\mathrm{g}^2 \mathrm{t}_i (\mathrm{a}(p)\mu_{X(p)} + 2\mu_{D(p)}) + 2\sigma_R^2}{\mathrm{t}_i^2 \mathrm{g}^2 \mathrm{a}(p)^2}. \quad (4.8)$$

In this equation, shot noise introduces a circular dependency between the estimate of $\mu_X(p)$ and the variances $\{\sigma^2_{X_i(p)}\}_{i \in T}$. The same applies for the

HDRI reconstruction

1. Acquire LDR images v_i and dark frames b_i
2. Assume constant variances $\sigma^{2\,(0)}_{X_i(p)}$
3. Estimate $\mu^{(k)}_{X(p)}$ assuming $\sigma^{2\,(k-1)}_{X_i(p)}$ (Eq. 4.5)
4. Estimate $\sigma^{2\,(k)}_{X_i(p)}$ assuming $\mu^{(k-1)}_{X(p)}$ (Eq. 4.8)
5. Iterate step 3, 4 until convergence (analogously for the dark current $\mu_{D(p)}$)
6. Smooth final $\mu_{X(p)}$ using the bandwidths derived from $\sigma^2_{X(p)}$ (see Sec. 4.6.2)

Figure 4.2: Pipeline for optimal HDRI reconstruction.

dark current. For this reason, the mean and the variances need be solved iteratively. Assuming initial constant variances, estimates for $\mu_X(p)$, $\mu_D(p)$ are obtained using Eq. 4.5. Then, the variances $\sigma^2_{X_i(p)}, \sigma^2_{D_i(p)}$ are estimated using Eq. 4.8. The estimation is iterated until convergence. The complete reconstruction pipeline is presented in Fig. 4.2.

4.4 Analysis of the Mitsunaga-Nayar Method

Mitsunaga and Nayar propose a weighting function where pixels values with higher signal-to-noise ratio receive higher weight. Given the exposure $E_i(p) = X_i(p)\mathrm{t}_i$ corresponding to a pixel value $V_i(p)$ in image i, they propose the weighting function

$$W_i^{\mathrm{MN}}(p) \equiv \frac{E_i(p)}{\sigma_{E_i(p)}} \approx \frac{E_i(p)}{\frac{\partial E_i(p)}{\partial V_i(p)}\sigma_{V_i(p)}}. \tag{4.9}$$

Furthermore, they assume that variance $\sigma_{V_i(p)}$ is constant across images with different exposure time, thus ignoring shot noise. This leads to the weighting

$$W_i^{\mathrm{MN}}(p) \approx \frac{E_i(p)}{\frac{\partial E_i(p)}{\partial V_i(p)}}. \tag{4.10}$$

For cameras with linear response (e.g. Eq. 2.19), the partial derivative $\frac{\partial E_i(p)}{\partial V_i(p)}$ is a constant. In addition, the irradiance $X_i(p)$ is also constant across images. Therefore, the weighting can be approximated as

$$W_i^{\mathrm{MN}}(p) \approx \mathrm{t}_i. \tag{4.11}$$

Now, let us compare this result the optimal weighting in Eq. 4.7. Under the assumption that the dark current is low (i.e., $\mu_{X(p)} \gg \mu_{D(p)}$), and that the readout noise is negligible in comparison to the shot noise (i.e., $\mathrm{g}^2\mathrm{t}_i\mu_{X(p)} \gg \sigma_R^2$), one can show that the Mitsunaga-Nayar weighting approximates the optimal weighting (i.e., $W_i^{\mathrm{MN}} \approx W_i^{\mathrm{opt}}$).

Table 4.2: Estimated sensor parameters for our two test cameras

Id	Model	ISO	g	L_0	$\hat{\sigma_R}^2$	L_{sat}
A	Canon EOS 5D	400	0.23	128	6.5	3709
B	Canon PowerShot S5	400	0.92	32	18	1023

However, if the shot noise component of $\sigma_{V_i(p)}$ is not neglected in Eq. 4.9, the resulting weighting can be approximated as

$$W_i^{\mathrm{MN'}} \approx \sqrt{t_i}, \tag{4.12}$$

assuming the same conditions under which W_i^{MN} is optimal. Therefore, this weighting no longer approximates the optimal function under any assumptions.

Two conclusions can be drawn from this analysis. First, the Mitsunaga-Nayar weighting works well in practice for situations where the readout noise and the dark current shot noise are low. This could explain its widespread use of this weighting in HDR applications. Second, their weighting function does not converge to the optimal weighting if the shot noise is not neglected. For this reason, it is not optimal to use weighting functions for HDR reconstruction that are proportional to the signal-to-ratio of the input images.

4.5 Experimental Evaluation

In this section, the performance of the proposed noise-optimal weighting function is compared with other weightings available in the literature. The comparison was performed on exposure sequences captured using two digital cameras: A Canon EOS 5D, 12-bit DAC (named *camera-A*), and a Canon PowerShot S5, 10-bit DAC (named *camera-B*), both set to ISO-400 gain factor, and with all noise removal features disabled. Camera parameters were estimated using one bias, one saturation, and 36 flat field frames, using the procedure described in Sec. 2.2.4. The resulting parameters are presented in Table 4.2. In order to assess the reliability of our camera model, the experiments were performed both on real world images and on simulated images. The simulation was performed using the ground truth irradiance (described below), and the calibrated camera parameters.

4.5.1 Ground Truth Acquisition

For each camera, a scene was setup such that the dynamic range spans at least four orders of magnitude. The scene was photographed using six different exposure times. In order to obtain a reference HDR image, 36 photographs and dark frames were acquired for each exposure time. Each set of 36 images was averaged into a single image, which produces a nominal

six-fold reduction in the camera noise. In addition, the sample variance was computed and projected to the irradiance domain using Eq. 2.24. The projected variance provides the ground truth irradiance variance per image $\sigma^{2(\mathrm{gt})}_{X_i(p)}$. Using the averaged images and their ground truth variances, the ground truth irradiance $\mu^{(\mathrm{gt})}_{X(p)}$ was derived using Eq. 4.5. Fig. 4.3b shows the ground truth irradiance for one of the scenes. Once the ground truth irradiance and its variance are computed, the maximum signal-to-noise ratio (SNR) can be estimated as

$$\mathrm{SNR}^{(\mathrm{gt})} = 20 \log_{10} \frac{\mu^{(\mathrm{gt})}_{X(p)}}{\sigma^{(\mathrm{gt})}_{X(p)}}. \tag{4.13}$$

Note that, in order to avoid the uncertainty introduced by shutter speed variability, the input images were normalized such that the spatial average is constant for every sample of each exposure time. Additionally, due to clamping the sample variance becomes unreliable as the the output values come close to the saturation limit. For this reason, the affected output values were excluded from the SNR estimation.

4.5.2 Performance Comparison

The quality of any given weighting function $w_i(p)$ depends on how well it emphasizes low variance samples, without completely discarding the information in the samples with higher variance. Given a single sequence of output values $\{V_i(p)\}_{i \in T}$, the variance of the resulting irradiance estimate is given by

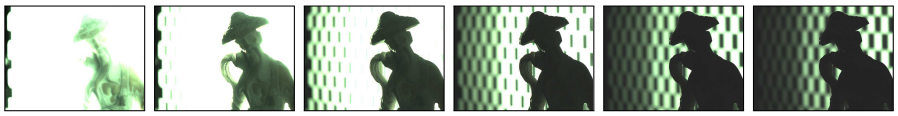
$$\sigma^{2(w)}_{\mu_{X(p)}} = \frac{\sum_{i \in T} w_i(p)^2 \sigma^{2(\mathrm{gt})}_{X_i(p)}}{\sum_{i \in T} w_i(p)^2}. \tag{4.14}$$

Note that for $w_i(p) = 1/\sigma^{2(\mathrm{gt})}_{X_i(p)}$, this expression is equivalent to Eq. 4.6, if the camera noise model is accurate. Given $\sigma^{2(\mathrm{gt})}_{X(p)}$, the ground truth irradiance $\mu^{(\mathrm{gt})}_{X(p)}$, it is possible to compute the SNR ratio achieved by a given weighting function $w_i(p)$ as

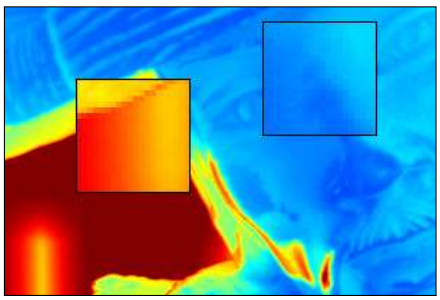
$$\mathrm{SNR}^{w} = 20 \log_{10} \frac{\mu^{(\mathrm{gt})}_{X(p)}}{\sigma^{(w)}_{\mu_{X(p)}}}. \tag{4.15}$$

In addition, the performance indicator should consider the bias error introduced by the reconstruction method. This includes errors caused by spatial noise sources such as DCNU and PRNU (defined in Sec. 2.2.2). The bias error is defined as

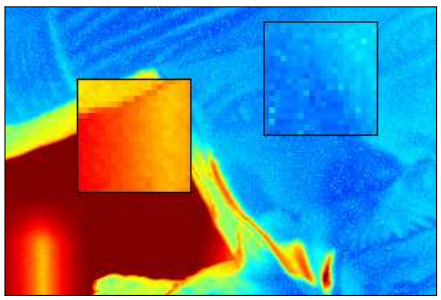
$$\mathrm{Bias}[\mu^{(\mathrm{gt,biased})}_{X(p)}] = \left| \mu^{(\mathrm{gt,biased})}_{X(p)} - \mu^{(\mathrm{gt})}_{X(p)} \right| \tag{4.16}$$



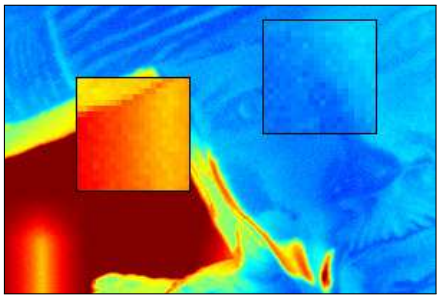
(a) Input image sequence



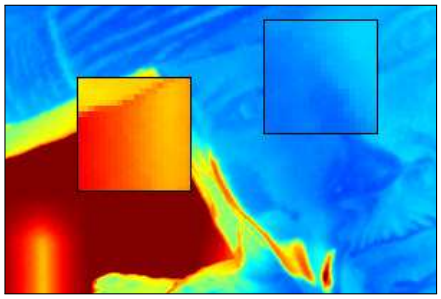
(b) Ground truth irradiance



(c) Using Mitsunaga-Nayar weighting



(d) Using proposed weighting



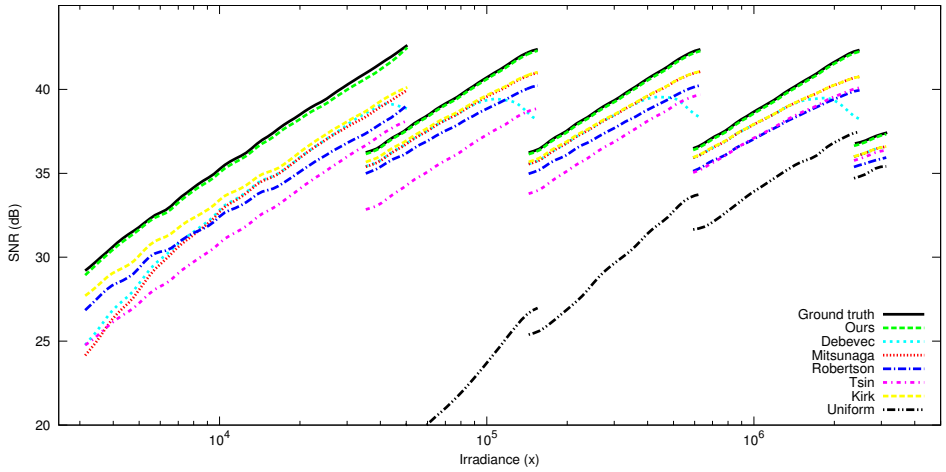
(e) Using proposed denoising

Figure 4.3: Example of HDR image reconstruction and denoising: (a) Ground truth HDR image, recovered from 180 images; only the green channel of the image is shown using a color code where red and blue denote high and low irradiances, respectively; (b) reconstruction using the Mitsunaga-Nayar weighting, which is sub-optimal at the lower irradiance range since it does not account for readout noise; (c) reconstruction using the proposed weighting based on the camera noise model; (d) denoised reconstruction using the predicted pixel variance to locally adapt the smoothing parameters to the noise of the image (see Sec. 4.6.2 for details on denoising).

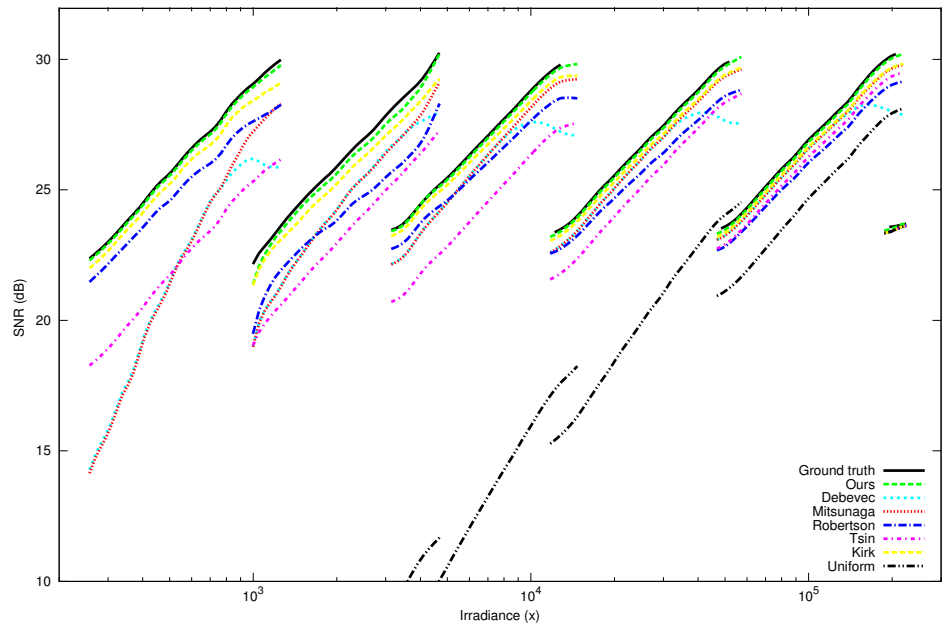
where $\mu_{X(p)}^{(\mathrm{gt,biased})}$ is corresponds to the same estimate $\mu_{X(p)}^{(\mathrm{gt})}$ but without accounting DCNU and PRNU during the reconstruction. Taking the bias error into account, the SNR estimate is given by

$$\mathrm{SNR}_{\mathrm{B}}^{w} = 20 \log_{10} \frac{\mu_{X(p)}^{(\mathrm{gt})}}{\sigma_{\mu_{X(p)}}^{(w)} + \mathrm{Bias}[\mu_{X(p)}^{(\mathrm{gt,biased})}]}. \tag{4.17}$$

Fig. 4.4 presents the SNR obtained by each weighting function on each

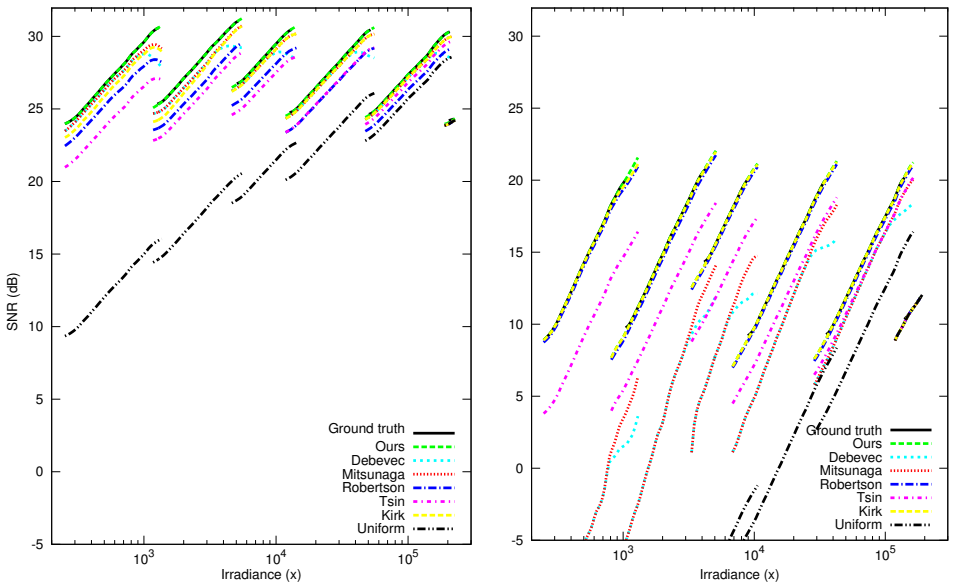


(a) *camera-A*, with readout noise σ_R^2



(b) *camera-B*, with readout noise $\sim 3\sigma_R^2$

Figure 4.4: Comparison of the signal-to-noise ratio (SNR) achieved by different weighting functions on cameras with different readout noise level. Note that the Mitsunaga-Nayar and Debevec-Malik weightings perform worse on the camera with higher readout noise (b) than in the one with lower readout noise (a) as these methods do not account for this noise source in their model. In contrast, our weighting achieves the highest SNR ratio among all methods, and it follows closely the maximum achievable SNR as given by the ground truth.



(a) $0\sigma_R^2$ (b) $10\sigma_R^2$

Figure 4.5: Signal-to-noise obtained on simulated cameras with extreme readout noise parameters: (a) Under zero readout noise, the Mitsunaga-Nayar weighting achieves the optimal SNR; the Debevec-Malik weighting achieves the optimal but only on half of the digital output range; (b) under high readout noise, the Robertson and Kirk-Andersen weightings approximate the optimal SNR; this occurs as the readout noise makes other noise sources negligible. The simulation was performed using the *camera-B* gain parameters.

of the two test cameras. Each SNR curve is obtained by fitting a non-parametric curve to the set of observed $(\mu_{X(p)}^{(\mathrm{gt})}, \mathrm{SNR}^{w}(p))$ for every pixel p. In the plot, the SNR for the upper irradiance range is virtually equal for all weightings; this occurs as there is just a single (non-saturated) image contributing to average, so the weighting has no influence in the final SNR. Conversely, the lower irradiance range is sampled by a larger number of (non-saturated) images, which are used to compute the irradiance estimate; in this case, the performance differences are more evident between different methods, since the resulting SNR depends on the specific weighting function. The discontinuities in the SNR lines occur at locations where the image with longest exposure becomes saturated. At that point, the image stops contributing to the irradiance estimate; this causes the sudden drop in the resulting SNR.

In all test cases, the SNR of the proposed weighting function closely follows the optimal SNR. It is also consistently higher than the SNR obtained with other weighting methods.

In the other hand, the uniform weighting achieves the lowest SNR, since samples with low and high variance contribute equally to the average. In [Bell08], this uniform weighting was proposed as the optimal weighting. That result was derived from a simulation where the noise was defined as additive zero-mean Gaussian noise with the same parameters for every image regardless of the exposure time. As explained in Sec. 2.2.3, this is not an accurate model for the camera noise.

The second best function corresponds to the Kirk-Andersen weighting [Kirk06]. This weighting properly accounts for temporal noise sources (except DCSN), but does not account for the spatial sources. Therefore, this method approximates the optimal SNR minus the bias error. Furthermore, the proposed weighting function has better confidence intervals that the Kirk-Andersen weighting (see Fig. 4.6). This follows from the type of irradiance estimates used during the variance estimation: The proposed weighting uses the averaged irradiance estimates (derived from Eq. 4.5), and the Kirk-Andersen weighting uses single-image irradiance estimates (derived from Eq. 2.23), which have higher variance than the former, hence the larger confidence intervals.

The weighting proposed by Tsin et al. [Tsin01] penalizes samples according to their standard deviation; for this reason, it overemphasizes values close to the noise floor, and gives less weight to values close to saturation, which have the lowest variance. The weighting by Debevec-Malik [Debevec97] also gives lower weight to values close to saturation; this explains the drop in the SNR that is occurs at the end of each segment, before the pixel values become saturated.

Similarly to the Kirk-Andersen weighting, all previous methods derive the weights from the digital output value of each individual image. This derivation causes the noise in the digital output to be transferred into the

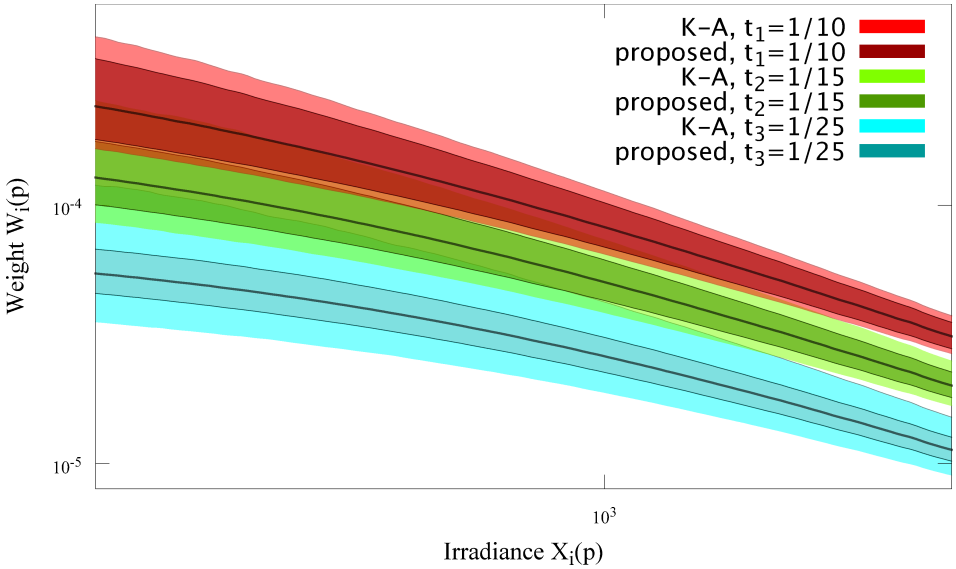


Figure 4.6: Comparison of the confidence intervals between the proposed noise-optimal weighting, and the Kirk-Andersen weighting. The proposed weighting function has better confidence intervals since it uses an irradiance estimate obtained from averaging all images in order to estimate the variance of a single image, as opposed to using a single image to estimate the same variance.

resulting weighting function. This transfer affects negatively all previous approaches, but becomes especially evident for Debevec-Malik and Mitsunaga-Nayar [Mitsunaga99] for pixels where the irradiance is low. The effect on the final HDR image is illustrated in Fig. 4.3c. The remaining weightings are less susceptible to this noise transfer since they are proportional to t_i^2, which reduces the influence of images with short exposure time (and higher variance).

Fig. 4.4 also shows a marked performance difference between the weightings of Mitsunaga-Nayar and Tsin et al., even though both assign weights according to the SNR of the digital output value. Mitsunaga and Nayar observe that the noise distribution is unknown, so they assume a constant distribution for all output values. As shown in Sec. 2.2.3, this assumption is invalidated by shot noise. Nevertheless, it can be shown (see Sec. 4.4) that this approximation leads to the maximum likelihood weighting when the readout noise approaches zero. This is illustrated in Fig. 4.5a, where the readout noise was suppressed using simulated images. When the readout noise is zero, the Debevec-Malik weighting also achieves an optimal SNR on the first half of the output range, where their assigned weights increase linearly.

Lastly, the weighting by Robertson et al. [Robertson03] performs consistently across the irradiance. This performance can be attributed to the t^2 factor contained in their weights, which heavily penalizes noisy values from shorter exposures. Still, they approximate the variance of the output values as derivative of the response function in log-scale (see Table 4.1). It can be shown that resulting weighting is proportional to t_i^3. For this reason, the images with longer exposure time receive overly high weights, so the optimal SNR is cannot be achieved. Nevertheless, there is a special case where this weighting achieves an optimal SNR. Fig. 4.5b illustrates this case using a simulated sensor that has high readout noise. In this case, all other noise sources become negligible, and any weighting that overemphasizes longer exposures will approximate the optimal weighting. This occurs as the effect of readout noise is lower when the exposure time increases. Since consumer imagining sensors often suffer from high readout noise, this could explain the popularity of the Robertson et al. method in publicly available HDR reconstruction software.

4.5.3 Gaussian Noise Assumption

A complete experimental validation requires testing the assumption of Gaussian distribution of the camera noise. The validation is performed on *camera-B* as it has a more representative readout noise component. The validation was performed using a sample of 36 images of a static scene. Three representative pixel locations were selected according to their expected output value: One right above the black level, one in the middle of range, and one

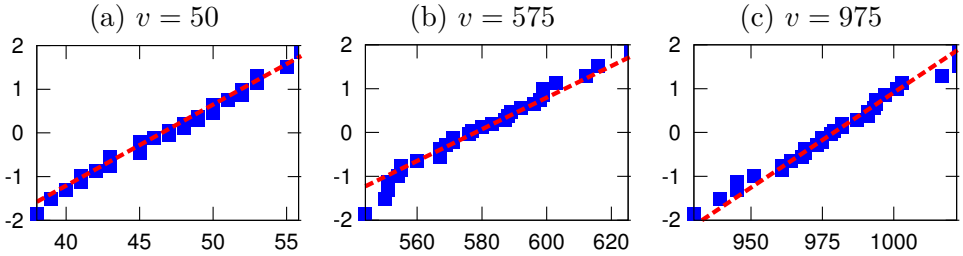


Figure 4.7: Validation of the Gaussian assumption on the noise distribution. Normal Q-Q plots were drawn for three samples of 36 pixel values on different pixel locations: (a) For an output value $v = 50$ just above the black level, (b) for an output $v = 575$ at the middle of the output range, and (c) for an output $v = 975$ just before saturation. Note that output range of the test camera used (*camera-B*) is $[32, 1023]$.

right before saturation occurs. Fig. 4.7 shows a Normal Q-Q plot for each of the three locations. The results indicate that the noise distribution follows a Gaussian distribution. This result is predicted by the camera noise model described in Sec. 2.2.3, since the sum of shot noise (Poisson distribution) and readout noise (Gaussian distribution) can be approximated by a Gaussian distribution, for large-enough irradiance values.

4.6 Further Applications of the Noise Model

The noise model applied to the reconstruction of optimal HDR images discussed in Sec. 4.3 has important practical implications for other HDR image processing tasks. In this section, two applications are presented: Exposure sequence planning (Sec. 4.6.1), and HDR image denoising (Sec. 4.6.1). In the first application, the noise model is used to pre-compute the sequence of exposure times required to best capture a given scene. This allows the user to capture higher quality HDR images using fewer exposures. In the second, since the noise model can be used to predict the noise of a reconstructed HDR image, it is possible to further improve their quality by performing a noise-aware image denoising process. This allows users to improve their images even after the HDR image has been captured and optimally-reconstructed.

4.6.1 Optimal Exposure Time Selection

The camera noise model presented in Sec. 2.2.3 can be applied to other HDR image processing tasks, besides HDR reconstruction. For instance, if a user needs to measure the irradiance of a scene with some minimum quality constraints (in terms of signal-to-noise ratio), the noise model can be used to devise an exposure sequence that satisfies such a requirement. Along

this line, this section presents a method for computing a set of exposure times $\mathbf{t} = \{t_1, t_2, \dots\}$ that, if used to capture a scene, it would produce an irradiance estimate with a guaranteed minimum SNR at every pixel location.

The proposed method makes two assumptions: The camera noise parameters are calibrated, and the irradiance distribution of the scene is known. For satisfying the first assumption, the calibration process described in Sec. 2.2.4 is used. Although not utilized in this work, Gallo et al. [Gallo12] proposed an algorithm for estimating the irradiance histogram of the scene using existing hardware in the camera.

Related Work

A few strategies have been proposed to generating optimal exposure sequences. Barakat et al. [Barakat08] points out that the average SNR of an HDR image increases with the number of images that are included in the average. In the other hand, Grossberg et al. [Grossberg03] explicitly set the desired sampling density of the radiance range. They obtain a sequence of exposure times by minimizing the difference between the desired sampling density, and the density achieved by the set of images. Their method is independent of the irradiance of the scene, and it allows to pre-compute tables of optimal exposure sequences that could, for instance, be included in the camera firmware. Following a different approach, Chen and El Gamal [Chen02] propose a method that explicitly tries to maximize the average SNR of the resulting HDR image. Their method provides optimal exposure sequences for the case where the irradiance distribution of the scene is uniform. For handing arbitrary scene distributions, they generalize this result by approximating the irradiance distribution using a piece-wise uniform function. The method proposed in this section follows a similar approach but with a different aim: To maximize the minimum SNR of the final HDR image.

Proposed Exposure Selection Method

An exposure sequence that achieves an specified minimum SNR could be obtained by optimizing the max-min function

$$\mathbf{t}^{\text{opt}} = \arg\max_{\mathbf{t}} \min_{p \in \mathcal{I}} \left(\frac{\mu_{X(p)}}{\sigma^{\mathbf{t}}_{X(p)}} \right), \tag{4.18}$$

where $\mu_{X(p)}(p)$ is the irradiance estimate at pixel p, and $\sigma^{\mathbf{t}}_{X(p)}$ corresponds to the expected variance of the irradiance estimate if it is reconstructed using the exposure sequence $\mathbf{t}$ (see Eq. 4.8). Note that this type of min-max function does not limit the number of images that are used to achieve the desired minimum SNR. Furthermore, it does not explicitly specify the target SNR. Nevertheless, a greedy algorithm can be applied to compute optimal sequences of increasing size until the target SNR is reached.

However, the sequences selected by such a greedy algorithm increase the SNR of the sequence slowly. This occurs as the only requirement is to raise the minimum SNR of the corresponding pixels as much as possible at each step. Instead, it is preferable to select images that increase the SNR for the most number of pixels. This can be achieved by optimizing the function

$$\mathbf{t}^{\text{opt}} = \arg\min_{\mathbf{t}} \mathop{\mathbb{S}\text{td}}_{p\in\mathcal{I}} \log\left(\frac{\mu_{X(p)}}{\sigma^{\mathbf{t}}_{X(p)}}\right), \tag{4.19}$$

which minimizes the standard deviation of the SNR in the image domain at each step. In this way, the differences in SNR between different pixel locations are minimized. This results in an even increase of the SNR across the irradiance range, since the SNR increases monotonically with the number of images [Barakat08].

Note that this method can also generate exposures sequences in cases where the irradiance of the scene $\mu_X(p)$ is unknown. For instance, the same method can be applied if only a hypothesis of the minimum and maximum irradiance of the scene is given. In this case, one can assume an uniform irradiance distribution over the range (or any other desired distribution), and minimize Eq. 4.19 using simulated images that follow the corresponding distribution.

Experimental Validation

For validation, a test scene (show in Fig. 4.3b) was selected as incoming irradiance. The target camera was simulated using the parameters of *camera-B*. The optimization algorithm was run with three SNR targets: 10dB, 20dB, and 30dB. The resulting exposure sequences have 2, 4, and 14 images, respectively. Fig. 4.8 shows the SNR obtained by each of the exposure sequences. The first image in the sequence (t_1 in the figure, ordered from top to bottom) corresponds to the longest exposure before saturation occurs. Subsequent images incrementally sample the irradiance range in order to increase the SNR of the region with the lowest ratio. The resulting exposure sequence samples the irradiance range in a way that the SNR peaks are evenly distributed. This distribution is achieved since the standard deviation of the SNR was chosen as minimization target.

4.6.2 HDR Image Denoising

In this section, I show that the camera noise model can be further applied to the problem of de-noising HDR images. If the variance of the irradiance estimate at every pixel is known, it can be used to smooth the noise in HDR images. By following this approach, only those scene features that fall below the camera noise level are smoothed, since the variance predicted by

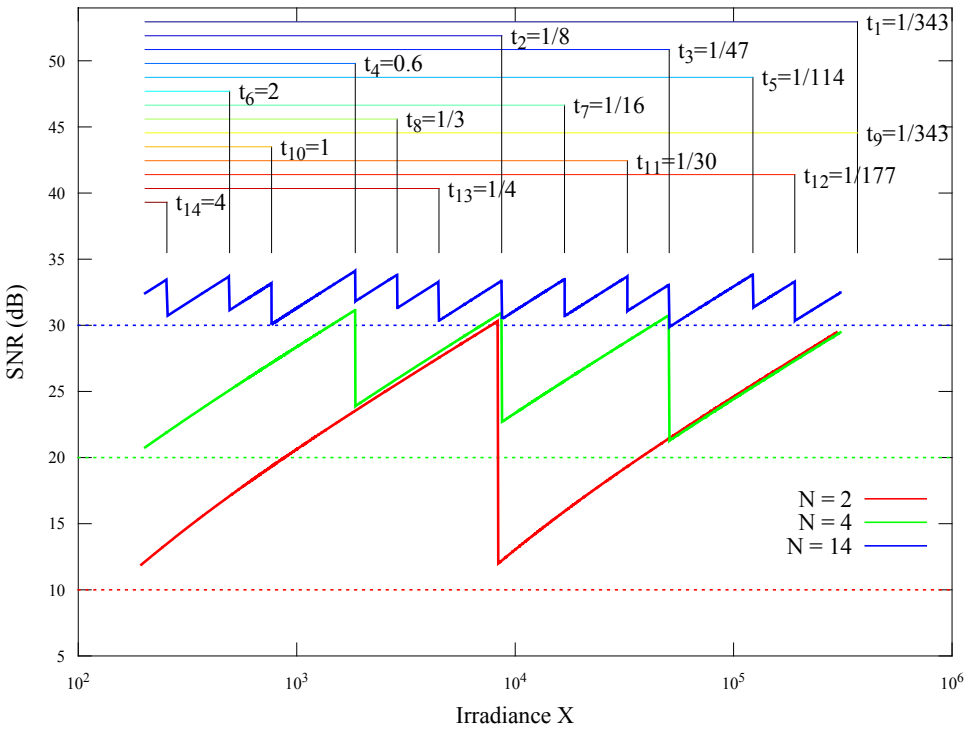


Figure 4.8: Optimal exposure sequences for a scene with known irradiance distribution: Three sets of exposure times were computed for obtaining a target SNR of 10dB (red), 20dB (green), and 30dB (blue), respectively. Each set required 4, 8, and 14 images, respectively, to achieve the target SNR. The simulation was performed using the parameters of *camera-B*. Peaks in the SNR plot correspond to the saturation point of individual images. The estimated exposure times are indicated at the peaks of their corresponding images.

the noise model only accounts for noise sources introduced during the image acquisition process.

This approach can be cast under the concept of *ideal spatial adaptation* described by Donoho and Johnstone [Donoho94]. In their paper, they discuss the advantages of having an *oracle* that provide information on how to best adapt an spatially variable function estimator such as a kernel smoother. This concept can be applied to image de-noising, where the function to be estimated corresponds to the undistorted image. If an oracle provides information about the noise level of a given pixel observation, methods such as wavelet shrinkage [Simoncelli96] or bilateral filtering [Tomasi98] can be used to remove noise in the image.

Following this approach, Lie et al. [Liu08] recover the noise level function (NLF) that predicts the noise standard deviation as a function of the image brightness. They estimate the upper bound of the true NLF as the lower envelop of the set of standard deviations computed within every segment in an image; the segments correspond to piece-wise smooth regions in the image, assuming an sparse image prior. They learn a prior on the shape of the NLF from simulated noisy images. The noise is generated using a database of response functions and a camera noise model. The resulting prior is used to regularize the NLF estimation from a single image, which serves as oracle for a subsequent denoising algorithm.

Proposed Method

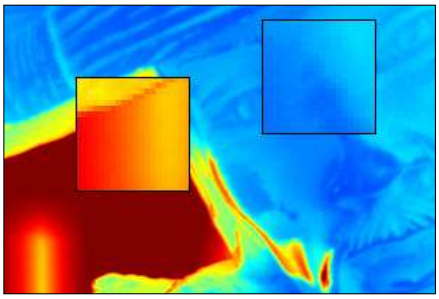
This section proposes a method that can estimate the noise level function with better accuracy, since the noise parameters are calibrated for the particular camera. Such noise level function can be applied to the problem of image HDR de-noising. For de-noising, the method of bilateral filtering [Tomasi98] is chosen. In this setting, the predicted noise level $\sigma_{X(p)}$ (computed using Eq. 4.8) can be used to set the bandwidth of the range kernel of the bilateral filter. Following this strategy, the de-noised image is given by

$$\mu^{\mathrm{bf}}_{X(p)} = \sum_{q \in \mathcal{I}} K_{\mathrm{space}}(p,q) K_{\mathrm{range}} \left(\frac{\mu_X(p) - \mu_X(q)}{\sigma_{X(p)}} \right) \mu_X(q) \tag{4.20}$$

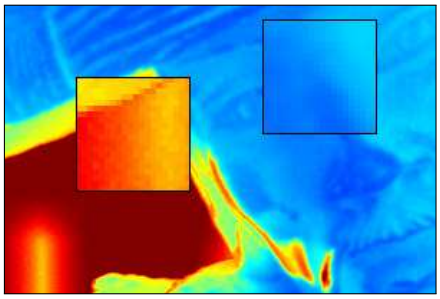
where K_{space} and K_{range} correspond to the kernels for penalizing the spatial distance and the value distance between two pixel p and q, respectively.

Experimental Validation

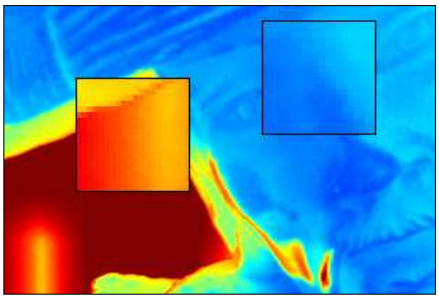
The proposed de-noising method is experimentally validated using the irradiance reconstruction shown in Fig. 4.9a, and its corresponding variance estimate. The resulting de-noised image is shown in Fig. 4.9c, where the



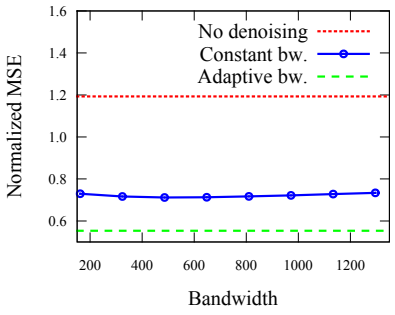
(a) Ground truth



(b) Best constant bandwidth



(c) Noise adaptive bandwidth



(d) Normalized MSE per bandwidth

Figure 4.9: Optimal bandwidth for HDR image denoising: (a) Ground truth HDR image, low irradiances shown in blue, high in red; (b) smoothing using the constant bandwidth with minimum normalized MSE; (c) adaptive smoothing according to the predicted noise level; (d) the normalized MSE shows that our adaptive method achieves lower error than any given constant bandwidth.

details in dark and bright regions are preserved despite the large differences in variance in both regions.

The performance differences between the proposed method and a baseline implementation of bilateral filtering with constant bandwidth are illustrated in Fig. 4.9b. The performance was evaluated as the mean normalized squared error (MNSE) between the de-noised irradiance and the ground truth. The baseline method was tested on a wide range of fixed bandwidths. The resulting bandwidth-MNSE plot for the baseline method (Fig. 4.9d) indicates that the denoising error obtained with the proposed noise-adaptive kernel bandwidth is lower than the error obtained with any constant bandwidth. The best constant bandwidth (450) is shown in Fig. 4.9b, where the details in the dark regions are visibly over-smoothed, and the noise in the bright regions is not fully removed.

To summarize, this section shows that the high frequency sensor noise visible in reconstructed HDR images can be successfully removed using bilateral filtering if the bandwidth of the range kernel is predicted using the variance of the signal, as estimated by the camera noise model.

4.7 Conclusion

This chapter presented a new pipeline for reconstructing noise-optimal HDR images. The input consists on a set of raw images acquired with different exposure times, and the output is a HDR image with maximal signal-to-noise ratio. The reconstruction pipeline is based on a very accurate camera noise model that takes into account the temporal and spatial noise sources that affect the camera acquisition process. This noise model enables the definition of an optimal weighting function for combining the measurements available on each input image into a single HDR value. The proposed reconstruction method incorporates the most complete camera noise model applied so far in the literature, and therefore, it achieves the highest signal-to-noise ratio among all currently available methods. This superior performance was demonstrated both theoretically and through experimental validation. In addition, the noise model was applied for improving additional HDR image processing tasks such as computing optimal exposure sequences and performing high quality HDRI denoising.

PART II

Editing of Video Sequences

The previous part proposed methods for editing image sequences acquired using digital cameras. The objective of such methods was to create low-noise ghost-free HDR images. The proposed methods apply non-parametric probabilistic inference tools in order to produce plausible results but without building models of the scene's content. Similarly, this part deals with editing video sequences, i.e., image sequences that are acquired using video cameras. These editing tasks can also benefit from the application of the non-parametric probabilistic inference methods that were used for HDR image editing.

In particular, this part deals with the problem of removing objects from video sequences. This problem is also called video inpainting, or video completion since the illusion of removing an object is achieved by inpainting or completing the appearance of the portion of the scene that was occluded by the object that needs to be removed. This can be achieved by borrowing the missing appearance from other parts of the video where it is visible.

For approaching the problem of object removal, one needs to consider two aspects of the class of input videos: The motion of the camera, and the motion of the remaining objects in the scene.

Concerning the camera motion, videos could be captured using static cameras (e.g. using a tripod), or with moving cameras (e.g. hand-held). On the other hand, the motion of the remaining objects in the scene can be static (e.g. the ground and buildings), or dynamic (e.g. people, cars, and trees). For reducing the complexity of the problem, the location of the object to be removed is assumed to be known, i.e., a video mask marking its location is given. This can be easily achieved using existing software. For this reason, the motion of the object to remove (e.g. static, dynamic) is considered irrelevant for the problem.

Now, let us analyze the problem from the perspective of the type of camera motion. If the camera is static, it could be possible to inpaint the remaining scene objects that are occluded. Such an inpainting can be performed if these objects are visible in a similar pose on other images of the video sequence. This is trivially true for static objects, and can often occur for dynamic objects with *redundant* motion. If this is the case, these views can be used as reference for the inpainting process. This type of inpainting problem will be addressed in Chapter 5.

In the second scenario, when the camera is moving, the appearance of the remaining objects on the reference views will suffer from perspective distortions. If these distortions are corrected, an inpainting algorithm can take advantage of the additional views of the occluded objects. In general, the distortion of these additional views can be corrected by rendering them from another viewpoint. This is possible whenever the camera location and depth for each of the images are known. These parameters can be estimated using multiple-geometry techniques that exploit the location of geometrical correspondences across frames. However, this estimation is challenging to

perform for dynamic scenes without building stronger models of the scene, since the location in space of these geometry correspondences is (naturally) not constant for moving objects. In addition, even when the scene is static, the estimation process is error-prone. With the aim of avoiding the estimation of depth and camera locations while at the same time keeping a minimal scene model, Chapter 6 presents an inpainting method for videos taken with moving cameras that is able to restore the appearance of static objects.

CHAPTER 5

Inpainting Dynamic Objects in Static Cameras

5.1 Introduction

To remove unwanted objects or artifacts from video sequences is a common task in television and movie production. A typical scenario occurs when the footage is taken in public locations, where it is often necessary to remove objects that accidentally enter the scene. Some objects may also have to be erased from a video sequence due to copyright issues, like advertisements or trademarks. Another scenario occurs when the film crew needs to be in the scene for technical reasons, and they need to be removed in post-processing. Additionally, production firms may need to restore damaged films, e.g., removing scratch lines and spots which are often observed in deteriorated film stock.

Removing undesired objects from video sequences implies *completing* or *inpainting* in a plausible way the appearance of the scene portion that was occluded by the undesired object (see Fig. 5.1 for an example). On images, this is a difficult problem that often requires manual interaction to achieve plausible results [Barnes09]. On videos, this difficulty is exacerbated due to the sensitivity of the human visual system to temporal artifacts [Wandell95]. Furthermore, it is common that in a given scene there are multiple moving objects each occluding or being occluded by the object to be removed. As a result, video completion is a tremendously difficult task that requires that artists spend many hours of tedious manual work for removing even small

Figure 5.1: Removing an object from a video sequence implies restoring the remaining dynamic objects (e.g. persons), and the background (e.g. ground, walls) behind the object: (top) Two frames from the input video sequence; (bottom) inpainted frames where the foremost person was removed using the proposed method.

objects. Consequently, many man-hours could be saved if this type of video completion tool would be available to the users. Besides, the availability of these tools could open new creative editing possibilities.

Still, building such an object removal tool is a challenging task: General video completion is an ill-posed problem, as there is no unique solution for completing the occluded regions. On the bright side, videos often contain a high degree of redundancy, with repetitive patterns occurring at different locations, times, and scales [Glasner09]. If available, this redundancy can be exploited to perform automatic video inpainting. Despite this potential advantage, there are very few scene-independent video inpainting methods proposed in the literature, and none has been demonstrated in real-world high-resolution scenes.

In this section, I present a video completion method that works by exploiting the redundancy in the video sequence. The proposed method is the first to demonstrate plausible inpaintings of dynamic objects in general scenes at high resolution. This method has been published in [Granados12b].

The proposed method assumes that the set of *missing* pixels to be completed is given as input. This set corresponds to the portion of the scene that is occluded by the object to be removed. This region can be easily marked using existing semi-automatic software [Bai09, Adobea].

In a nutshell, video completion is performed by locating other (partial)

views of the occluded objects in the missing region, and constructing a composite of these views for filling it, in way that the resulting completion looks plausible. This is performed by computing offsets from the set of missing pixels to other pixels in the video. Each offset points to the *source* pixel whose color will be used to inpaint the missing pixel. A Markov Random field prior is assumed over the offset field. The prior potential makes sure that the resulting inpainting is plausible, i.e., that the resulting offsets lead to pair-wise compatible colors between adjacent pixels. The resulting energy function is presented in Sec. 5.3.

The proposed algorithm builds upon the closely related concepts of *correspondence maps* [Demanet03], and *shift maps* [Pritch09], which were proposed for image completion, and image retargeting and reshuffling, respectively. In these methods, the desired image is obtained by computing an offset field that minimizes the dissimilarity between *patches* around the pixels in the missing region, and the corresponding source patches outside the missing region. A related concept, the bi-directional similarity function proposed for visual summarization [Simakov08], includes a coherence term that assign a high cost to those patches in the resulting image that are not coherent with (i.e., not found in) the original image. Nevertheless, Sec. 5.4.2 shows that a simple extension of the concept of correspondence maps to the video domain does not lead to satisfactory results. The proposed method achieves better results by carefully adapting the energy function to the case of video volumes, ensuring that the optimization produces results that are both spatially and temporally coherent. These new developments are discussed in Sec. 5.4.2.

The proposed algorithm is evaluated using several high-resolution videos of challenging scenes (Sec. 5.4). These scenes feature multiple occlusions, and non-trivial motions, where the high resolution makes any potential mistakes easily noticeable. The inpainting results are compared against state-of-the-art methods on such scenes. The results of the comparison show that the proposed method consistently produces better inpaintings that other competing methods.

In addition, I propose an interface for making effective use of user input. This user input can help the inpainting algorithm in two ways. First, it can reduce computation time drastically if the space of possible offsets is constraint by providing tracks for the dynamic objects to be inpainted (Sec. 5.3.3). Second, the user can refine the inpainted result by providing the rough location of a good source for an specific missing region. This refinement can be used for correcting situations where the automatic completion fails (Sec. 5.4.4).

5.2 Previous Work

The existing video completion algorithms can be broadly classified into two categories: Object-based methods, and patch-based methods. In general, *object-based* methods segment the occluded objects in order to construct an appearance model using other unoccluded views found in the video. The purpose of such a model is to predict the appearance of the object during the occlusion (i.e., inside the missing region). Although such models can make very plausible predictions after constructed, they require accurate object segmentations. In addition, the space of motions that can be modeled is often restricted to periodic motions. Therefore, this category of methods is less applicable in practice.

On the other hand, patch-based methods do not make strong assumptions about the type or extent of the objects found in the scene. Instead, the fundamental unit of comparison is a *patch*, i.e., a contiguous subset of pixels in the video volume, which do not have to be defined in correspondence with the objects in the scene. These methods perform inpainting by finding a suitable arrangement of patches sampled from different locations of the video. These methods can be further classified into local and global methods.

Local and global methods differ in the mechanism used for constructing the arrangement of patches that fills the missing region. *Local* methods take a recursive approach, where the hole is filled incrementally by finding suitable patches starting from the boundary of the hole, in a way that the missing region is reduced at every step of the algorithm. This strategy generally results in faster algorithms than global methods. However, incremental filling does not guarantee global consistency, and therefore, these methods are less suitable for large spatio-temporal holes. Instead, *global* methods define a consistency measure which is typically represented as an energy functional. If defined properly, the minima of the energy should correspond to arrangement of patches that produce plausible inpainting solutions. These methods can be applied even when the missing region or *hole* is large in space and in time, since the applied optimization procedures distribute the error across the missing region. However, finding optima for such global energies is unfeasible due to its high computational complexity, and therefore, only local minima can be obtained. In the following sections, each of these categories is discussed in more detail.

5.2.1 Object-based Methods

Object-based inpainting methods rely on the possibility to construct models for the occluded objects. These models are built using additional information about these objects such as accurate segmentation, layer decomposition,

and motion estimation. For inpainting dynamic objects, these methods often make stronger assumptions about the motions, such as periodicity.

The method of Venkatesh et al. [Venkatesh09] requires a segmentation of the occluded object. Using this segmentation, a database of segmented frames is constructed, including only those frames where the occluded object is fully visible. Using dynamic programming, the holes are completed by aligning frames in the database to the partially or fully occluded frames in the missing region. This method requires that the segmentation be very accurate, and that the motion be mostly cyclical for each occluded object.

This idea is extended by Ling et al. [Ling09], where the contours of the object of interest are estimated using motion information. These contours are used to retrieve the relevant frames from the database using an approach similar to Venkatesh et al. [Venkatesh09]. In order to make the query process robust to posture differences, the query postures are synthesized based on local segments of the object.

The technique by Jia et al. [Jia06] assumes a periodic motion of the occluded objects. Their method segments the video into background and foreground, and proceeds to inpaint each separately, followed by an integration process. For inpainting the background, a layer decomposition is performed with user assistance, followed by layer projection onto the frames with missing regions. For inpainting the occluded dynamic objects, they warp and align the trajectory of each occluded object along the missing region with the reference trajectories outside. This is possible as the motion is assumed to be periodic.

Object-based methods can produce plausible completions in several scenarios. However, these scenarios are restricted to particular classes of motions, e.g., periodic motion [Venkatesh09, Jia06], or require the motion to be simple enough such that it is feasible to densely sample set of postures of each object [Ling09]. Furthermore, to take advantage of the model assumptions, the completion of dynamic objects and background needs to be performed independently. This separation requires an accurate segmentation of the background, and a final merging step that can introduce an unnatural appearance to the final result. In contrast, the local and global patch-based methods described in the next sections do not take advantage of object-based priors, but consequently, do not suffer from these limitations.

5.2.2 Local Methods

The method of Patwardhan et al. [Patwardhan05] falls within the category of local, patch-based methods. This method inpaints the missing pixels in a sequential order. The order is given by a priority measure based on the amount of non-hole pixels around the missing pixel, and on the presence of structure (e.g. edges, motion boundaries) around the pixel. In the order of highest priority, the algorithm copies the patch that best match the

neighborhood of each missing pixel, until all pixels are filled.

This algorithm was later improved to handle camera motions parallel to the image plane [Patwardhan07], and to handle general camera motions and reduce temporal discontinuities [Shih09]. In general, local methods are faster than global methods (described next), but the resulting completion is not guaranteed to be globally coherent.

5.2.3 Global Methods

The seminal paper of Wexler et al. [Wexler07] falls within the category of global methods. Their method starts by gathering a set of spatio-temporal patches centered at every non-hole pixel of the input video. The collection of patches constitutes a database that reflects the local appearance of the video. From this database, the patch with the closest color is selected for inpainting each missing pixel, following an iterative algorithm. This iterative selection decreases the energy of global function that penalizes the disagreement between the source patches that are adjacent in the missing region. Since the adjacent patches overlap, there are several candidate colors for filling each missing pixel. A final color value is obtained as a linear combination of the candidates colors.

The method of Shen et al. [Shen06] tries to retain the advantages of global approaches while at the same time reducing computational complexity. For this purpose, they track every pixel of the occluded object throughout the video such that, during the energy minimization stage, the search space for each pixel is reduced from a 3-dimensional to a 2-dimensional manifold. Still, this simplification comes at the expense that only objects undergoing pure translations or periodic motions can be handled.

A more indirect approach is known as *motion transfer*. The idea is to compute a motion field that is used to propagate pixel colors from outside the missing region. The motion field can be computed, for instance, by gradually propagating motion vectors [Matsushita06], or by computing motion patch similarities [Shiratori06]. However, methods based on motion transfer allow the completion of only a relatively small number of frames. Completion of large time intervals is more challenging as the pixel propagation process suffers from smoothing artifacts.

5.2.4 Offset-based Global methods

The method proposed in this chapter falls within the category of global methods, and it is based on the shift-map image editing framework [Pritch09]. The objective of this framework is to compute a offset field or *shift-map* over the image domain that corresponds to a solution of one of several image editing tasks, including inpainting, reshuffling, and re-targeting. This vector field determines an offset from every pixel in the resulting image to

the location in the original image from where it should take its color value. For the particular problem of image completion, they constrain the offsets assigned to missing pixels such that they point to pixels outside the hole. The offsets are computed by minimizing an energy function that penalizes the discrepancy between the sources of adjacent pixels.

Hu and Rajan [Hu10] apply the concept of shift maps to video sequences for the problem of video retargeting. For the retargeting, an offset field is estimated from the original video domain to a domain of different resolution. This offset field is obtained by minimizing an energy function that penalizes the spatial and temporal discrepancy between the sources of adjacent pixels. Note that offsets along the time axis are not required for retargeting.

However, solving the video completion problem by simply extending shift-maps to allow temporal offsets does not produce plausible results. This occurs as the spatial and temporal dimensions have fundamentally different properties (Sec. 5.4.3). The proposed method defines an energy function that is designed to account for these differences. This energy is described next.

5.3 Video Inpainting Method

The input required by the proposed video inpainting algorithm is two-fold: The video sequence to be inpainted, and a mask that determines the pixels to be completed, which correspond to the object to be removed. The first input is a color video sequence $\mathbf{V} : \mathcal{V} \mapsto [0, v_{\max}]^3$, where $\mathcal{V}$ is the 3-dimensional video domain, and $v_{\max}$ is the maximum digital output value of the camera (usually 255, for 8-bit video). This video sequence can be thought of as a *video volume*, where frames are stacked along the temporal dimension, and each pixel color $\mathbf{V}(x, y, t)$ is indexed using the spatial coordinates x, y and the frame index t.

The second input is a mask $\mathbf{M}_R$ that defines the set of pixels $\Omega = \{p \in \mathcal{V} : \mathbf{M}_R(p) = 1\}$ that correspond to the spatio-temporal hole in the video volume left by the object to be removed. For constructing this mask, any video segmentation algorithm can be used; the implementation of [Bai09] available in Adobe After Effects [Adobea] was chosen for this purpose.

The objective of video inpainting is to determine a substitute color for the pixels in Ω, in a way that the object to be removed is not shown, and the final video looks plausible to humans. To achieve this goal, the strategy of the proposed method is to fill the color of each missing pixel in Ω by computing an *offset* that points to a suitable location where the pixel color can be copied. The suitability of such offsets is encoded in an energy functional, which is described next.

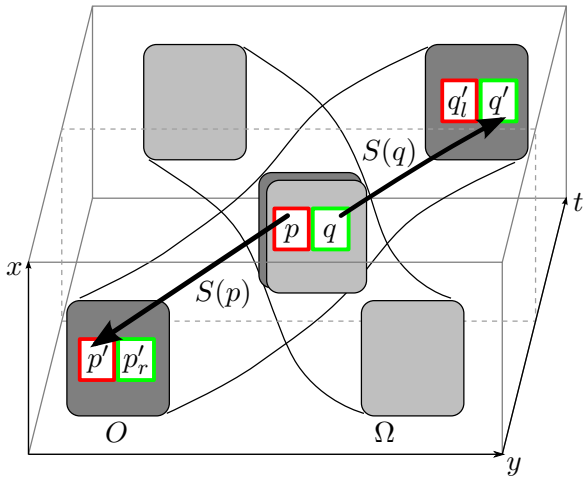


Figure 5.2: Video volume inpainting. A pair of missing pixels p, q inside the spatio-temporal hole Ω can be plausibly filled by pixels $p' = p + S(p)$, $q' = q + S(q)$ if the appearance for the neighbors of p' and q' is consistent. This consistency is measured by the color and gradient difference between the colors of the pairs (p', q'_l) and (p'_r, q'). For an inpainting to be plausible, such consistency has to be achieved for all pairs of adjacent pixels in Ω.

5.3.1 Energy Functional

Formally, the proposed inpainting method is defined as follows. Given an input video I, and a set of missing pixels Ω, an offset $S(p) = (d_x, d_y, d_t)$ is computed for every missing pixel $p \in \Omega$. The resulting offset set is called an *offset volume*. Using this offset volume, the inpainted video $\mathbf{V}'$ is constructed by assigning to each missing pixel the color of its offset location outside the hole, i.e.

$$\mathbf{V}'(p) = \mathbf{V}\left(p + S(p)\right), \text{ for } p \in \Omega, \text{ and } p + S(p) \in \Phi, \tag{5.1}$$

where $\Phi = \mathcal{V} \setminus \Omega$ denotes the unoccluded part of the video volume. This is illustrated in Fig. 5.2.

The inpainted video is required to be plausible to humans. However, without high level scene understanding, this plausibility constraint can be only approximated using low-level cues. Therefore, the proposed method achieves plausibility using two complementary low-level constraints. First, the offset volume should be as *coherent* as possible, i.e., large contiguous portions of the video should be copied whenever possible. The rationale behind this approximation derives from the observation that the unoccluded part of the videos is self-consistent, and therefore, large contiguous portions of them are also self-consistent. Second, the boundaries between adjacent coherent regions in the offset volume should be consistent, i.e. the sources

of the adjacent regions should have similar appearance, so the boundary is not noticeable by humans. These two plausibility constraints are encoded in the energy functional

$$\mathcal{E}(S) = \sum_{(p,q)\in\mathcal{N}\wedge p\in\Omega} \mathbb{1}_{\{S(p)\neq S(q)\}} V_{p,q}\left(S(p), S(q)\right), \tag{5.2}$$

where $\mathcal{N}$ denotes the set of adjacent pixels in a 26-neighborhood system on the video volume.

In the energy, the indicator function $\mathbb{1}_{\{S(p)\neq S(q)\}}$ satisfies the first constrain. This function assigns a zero cost to adjacent pixels p, q whenever their corresponding offsets are the same. This assignment decreases the cost of offset volumes that contain large contiguous regions of constant offset. On the other hand, the prior potential $V_{p,q}$ satisfies the second constraint. It assigns a high cost to boundaries or discontinuities in the offset volume that are not consistent. This consistency is measured as the color and gradient difference between the two sources of adjacent pixels p, q, whenever their offsets differ. This prior potential is defined as

$$\begin{aligned} V_{p,q}(S(p), S(q)) = \; & \tau(p,q)\gamma(p,q)\cdot \\ & \Big[\Big(\| \mathbf{V}(p+S(p)) - \mathbf{V}(p+S(q)) \|_2^2 + \\ & \quad \| \mathbf{V}(q+S(p)) - \mathbf{V}(q+S(q)) \|_2^2 \Big)^{\psi} + \\ & \beta \Big(\| \nabla\mathbf{V}(p+S(p)) - \nabla\mathbf{V}(p+S(q)) \|_2^2 + \\ & \quad \| \nabla\mathbf{V}(q+S(p)) - \nabla\mathbf{V}(q+S(q)) \|_2^2 \Big)^{\psi} + \lambda \Big], \end{aligned} \tag{5.3}$$

where τ is a weighting function that balances the importance of spatial and temporal coherence, γ is a weighting function that increases the importance of inconsistencies close the hole boundary, β is a scalar that determines the importance of gradient inconsistencies with respect of color inconsistencies, ψ determines the type of penalizer to be used, and the scalar λ corresponds to the Potts model [Potts52]. The criteria applied for selecting these parameters are discussed next.

Parameter Selection of β, ψ, λ

First, the parameter β is set such that the color and gradient differences have equal contribution to the energy. To achieve this, the value is fixed to $\beta = \frac{1}{2\sqrt{2}}$. This value is justified as the range of gradient differences is twice as large as the range of the color differences, hence the factor $\frac{1}{2}$. The variance of the difference induced by shot noise only (see Sec. 2.2.2), assuming linear camera response, is also twice as large, hence the factor $\frac{1}{\sqrt{2}}$.

The proper choices of the remaining scalars ψ and λ are crucial for the performance of the inpainting algorithm. The value of the exponent ψ in

Eq. (5.3) is fixed to $\frac{1}{2}$, such that the resulting potential is more tolerant to outliers. This design differs from the L_2 square regularizer originally proposed in [Pritch09], which is chosen, according to the authors, to produce less coherent offset maps. In addition, the value chosen for this parameter ensures that the resulting prior potential $V_{p,q}$ is a metric, as in that case (i.e., $\psi = \frac{1}{2}$), the resulting potential corresponds to the Euclidean norm of the vector containing the color and gradient differences between the two source locations on each color channel. When $V_{p,q}$ is a metric, the resulting energy function is sub-modular. This implies that an optimization process based on graph cuts generates solutions whose energy is within a factor of the global minima (see Sec. 2.1). In the proposed energy, the maximum factor corresponds to the value of the scalar $\lambda = 0.1$. In contrast, the energy proposed in [Pritch09] is designed to be not sub-modular.

Design of the Weighting Function $\gamma(p, q)$

The algorithm is required to produce inpaintings that are consistent with the boundary of the missing region, and that are also consistent inside the missing region. Therefore, the purpose of the weighting function γ is to assign the same importance to achieving consistency at the boundary of the missing region Ω, as to achieving consistency inside the missing region.

In order to illustrate this objective, let us assume that an uniform weight γ is applied to the pair-wise potentials $V_{p,q}$ across the offset volume. Please observe that there is a large difference between the number of missing pixels at the boundary of the hole, and the number of pixels completely inside the hole volume. Given this situation, the total cost of the inconsistencies occurring inside the hole dominate the total cost of the inconsistencies occurring at the boundary, even assuming equal inconsistency everywhere in the volume. For this reason, a constant offset volume, i.e., one that assigns the same offset to every missing pixel, can have lower cost than other offset volumes that have less uniform offsets but are more consistent with the boundary of the missing region. This occurs as in Eq. 5.2 the consistency cost of a contiguous region with constant offset is only evaluated at its boundary.

In order to avoid this situation, the weighting γ is defined such that the weight $\gamma(p, q)$ is *sufficiently* larger than $\gamma(r, s)$ whenever the *hole depth* of a pair (p, q) is smaller than the depth of pair (r, s). The hole depth $d(p, q)$ is defined as

$$d(p, q) = \frac{1}{2}[d_C(p, \partial\Omega) + d_C(q, \partial\Omega)], \tag{5.4}$$

where the distance $d_C(A, \mathcal{B})$ between a point A and a set $\mathcal{B}$ is defined as the minimum of the *Chebyshev* distance[1] in a 26-neighborhood system between

[1] Also known as the chessboard distance, the Chebyshev distance between two points corresponds to the maximum distance along any dimension.

A and the elements of $\mathcal{B}$.

Based on the hole depth $d(p,q)$, the set of adjacent pixels $\{(p,q)\}$ with $p,q \in \Omega$ is partitioned such that each partition P_i consists of pairs that have the same distance to the boundary of the hole. In addition, the partitions are ordered in increasing distance, i.e., if $d(p,q) < d(r,s)$ for $(p,q) \in P_i$ and $(r,s) \in P_j$ then $i < j$.

If the weighting function γ is defined such that the total weight for the pairs in P_i is twice the total weight of the pairs in P_{i+1}, i.e.

$$\sum_{(p,q)\in P_i} \gamma(P_i) = 2 \sum_{(r,s)\in P_{i+1}} \gamma(P_{i+1}), \tag{5.5}$$

the following weighting function is obtained

$$\gamma(P_i) = \frac{2|P_{i+1}|\gamma(P_{i+1})}{|P_i|}, \tag{5.6}$$

where $|P_i|$ is the number of pairs in P_i, and the cost of the n-th partition that is farthest from the boundary is defined as $\gamma(P_n) = 1$. Using a geometric series approximation it can be shown that

$$|P_i|\gamma(P_i) \approx \sum_{i<j\leq n} |P_j|\gamma(P_j), \tag{5.7}$$

which is the desired behavior of the weighting function.

Intuitively, this weighting can be understood as field guiding the direction of information flow during the inpainting process, i.e., from the boundary of the missing region where a reliable visual context is available, into its interior where no context exists. A similar weighting design is presented in [Wexler07], and the particular differences are discussed in Sec. 5.4.2.

Design of the Space-time Weighting $\tau(p,q)$

The weighting τ accounts for the differences in importance between spatial and temporal inconsistencies. It is defined as

$$\tau(p,q) = \begin{cases} \alpha & \text{if } p - q = (0,0,\pm 1) \\ 1 & \text{otherwise,} \end{cases} \tag{5.8}$$

where α is a constant that controls the importance of incoherencies occurring along temporally adjacent pixels. If a 26-neighborhood system is used, the scalar α is set to $\frac{8}{18}$. This factor corresponds to the ratio between the number of neighbors in the same frame (8) and the number of neighbors in adjacent frames (18). This reflects the fact that the temporal and spatial coherence should have equal importance in order to achieve plausible inpaintings.

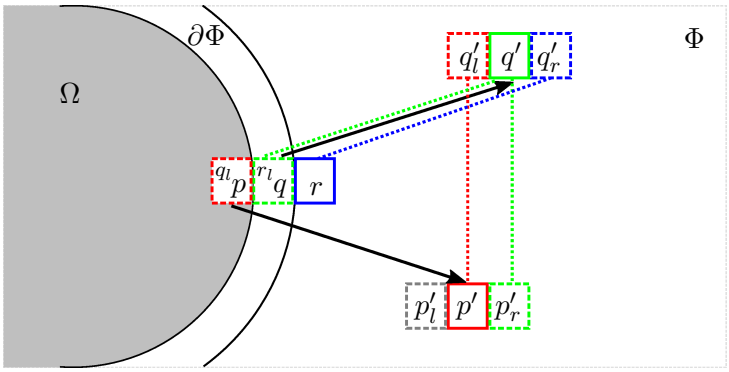


Figure 5.3: In addition to finding offsets for missing pixels $p \in \Omega$, the proposed method finds offsets for boundary pixels $q \in \partial\Phi$, although they are not occluded by the object to be removed. These offsets are computed so that the neighbors of q inside the hole, e.g. q_l, are always defined. In the figure, dotted lines denote the set of pixel color differences involving q that are evaluated in Eq. 5.2, namely $\mathcal{V}_{p,q}(S(p), S(q))$ and $\mathcal{V}_{q,r}(S(q), \mathbf{0})$.

Handling of Boundary Pixels

The pixels adjacent to the boundary of the hole require a special treatment. As defined in Eq. 5.3, $V_{p,q}$ is undetermined for pairs of pixels (p, q) where only p is a missing pixel, i.e., $p \in \Omega$ and $q \in \partial\Phi$. Since q is outside the hole, no offset is estimated for that location, or equivalently, the pixel can be considered to have a zero offset $S(q) = \mathbf{0}$. In this case, the pixel color $\mathbf{V}(p + S(q)) = \mathbf{V}(p)$ in Eq. 5.3 is drawn from inside the hole, which is an invalid color. This problem can be solved by estimating offsets for the boundary pixels $q \in \partial\Phi$ during the optimization procedure, in a way that Eq. 5.3 becomes well defined. This is illustrated in Fig. 5.3.

5.3.2 Multi-Resolution Optimization

The energy defined in Eq. 5.2 is a discrete non-convex functional, and finding a global minimum is a NP-hard problem [Veksler99]. Instead, a local minima is computed using the expansion move algorithm and graph cuts (see Sec. 2.1.2). In the proposed functional, the set of unknowns corresponds to the offset assigned to each missing pixel $p \in \Omega \cup \partial\Phi$, and the set of labels corresponds to the possible offsets starting from a missing pixel and pointing to other unoccluded pixels within the video volume.

Still, minimizing this energy functional using graph-cuts is challenging due to the very large size of the label set. To tackle this problem, we take a multi-resolution approach [Pritch09]: A video pyramid is constructed by reducing the spatial resolution by half until the resulting label set size

allows the optimization of Eq. 5.2; the particular criteria is described below. Masks are down-sampled in a conservative way such that missing pixels in finer levels remain as such in coarser levels. No down-sampling is performed along the time axis as this could introduce temporal discontinuities.

The offset volume is first optimized for the coarsest pyramid level, and it is subsequently up-sampled as an initial guess for the next (finer) pyramid level using nearest neighbors interpolation; the offset magnitudes are doubled to match the higher resolution. This process is repeated until the original resolution is reached.

On the coarsest pyramid level, the size of the label set is $(\frac{2w}{2^k} - 1)(\frac{2h}{2^k} - 1)(2t-1) \approx \frac{4wht}{(2^k)^2}$, where w, h, t are the width, height, and length of the video, respectively, and k is the number of levels in the pyramid. The number of levels is set such that the number of pixels in the missing region is smaller than 100^3. In this way, the optimization procedure remains feasible on standard computing hardware. For the remaining finer pyramid levels, only small offset adjustments relative to the initial estimate are examined, i.e., the label set is restricted to offsets with value $\{-1, 0, 1\}$ on each coordinate.

Run-time Complexity

The worst case run-time complexity of the proposed algorithm is $O(n^3N)$, where n is the total number of missing pixels in the coarsest pyramid level, and N is the size of the label set in the same level. Assuming that the shape of the video volume is cubic with side length l, the size of the label set can be approximated as $N \approx 4l^3$, which results in a worst case run-time complexity of $O(n^3l^3)$. Due to this cubic increase with the input size, it is extremely important to keep both the number of missing pixels and label set size as small as possible. This can be achieved by splitting the inpainting problem into several smaller sub-problems. This process is described next.

5.3.3 User-Assisted Reduction of Label Space

To make the optimization feasible on high-resolution videos, the number of missing pixels, and the label set size of the energy functional (Eq. 5.2) need to be reduced as much as possible. To achieve this goal, in addition to multi-resolution optimization, the inpainting problem is divided in two stages: Inpainting of the stationary objects, and inpainting of the dynamic objects. Stationary objects are defined as objects that do not change their global position in space but can change their local appearance between frames. This type of objects includes static background undergoing illumination changes. On the other hand, dynamic objects can change both their position and appearance between frames. In the first stage, stationary objects are inpainted, followed by the inpainting of dynamic objects.

For the first stage, the size of the label set can be reduced to $(2t-1)$ as only temporal offsets are required. This step is performed only once for filling all missing pixels in the video volume with their corresponding background.

In the second stage, each of the occluded dynamic objects is inpainted independently. Unlike in the first stage, inpainting of dynamic objects requires using the full set of possible offsets within the video volume. The label set for each dynamic object is reduced by restricting the set of possible offsets to lie within a tight video volume centered on the trajectory of the object [Jia05]. Given that humans can discern better the location of objects in video sequences than automatic tracking methods, especially in crowded scenes, I propose an interface to quickly specify the trajectories of occluded objects. Using the interface, the user can provide a tight video volume for each occluded dynamic object. This volume is given by two 2-dimensional masks defined over a xt- and a ty-projection of the video volume in the plane (Fig. 5.4). The set of pixels inside of the tight video volume are used to constrain the set of target pixels that need to be reachable from each missing pixel. Please note that even when dynamic objects are inpainted independently, the background and foreground within their corresponding bounding boxes are still inpainted simultaneously, i.e. there is no separation between the background and foreground inpainting processes. This user-guided tracking is applied in all results presented in Sec. 5.4, except for the *beach-umbrella* and *duo* sequences.

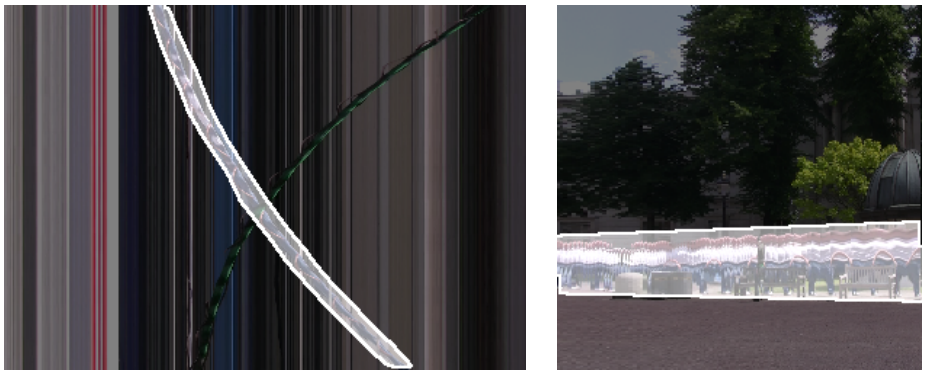
In addition, the proposed method prunes from the label set those offsets that only point to irrelevant background regions. These regions are detected by performing foreground thresholding using an estimated background model [Granados08].

5.4 Experimental Validation

This section presents the experimental validation of the proposed method using real-world high-resolution video sequences (Sec. 5.4.1), followed by a comparison with the most closely related methods from the literature, i.e., the inpainting method of Wexler et al. [Wexler07], and a straightforward extension of the method of Pritch et al. [Pritch09] to video volumes (Sec. 5.4.2). Additionally, an experimental validation of the parameters selection is presented in Sec. 5.4.3, and an interface for user-guided refinement of the inpainting results is proposed in Sec. 5.4.4.

5.4.1 Inpainting Results in Test Sequences

The proposed algorithm is validated on six video sequences corresponding to four different scenes. These sequences will referred to as *beach-umbrella*



(a) xt projection (b) ty projection

Figure 5.4: Interface for tracking occluded objects: To reduce the computation time, the space of possible offsets is restricted to cover only the region spanned by each occluded object. The figure highlights the mask drawn by the user on top of a xt projection and a ty projection of an input video.

(Fig. 5.8–top), *park-groundtruth* (Fig. 5.8–middle), *park-simple* (Fig. 5.8–bottom), *duo* (Fig. 5.5), *park-complex* (Fig. 5.6), and *museum* (Fig. 5.7). The *beach-umbrella* sequence was introduced by [Wexler07]. This sequence has a comparably low resolution with respect to the remaining sequences presented in this section, which were shot in Full HD resolution (anamorphic 1440×1080). The resolution of the input video sequences is a relevant factor in the evaluation of video inpainting methods, since artifacts become more noticeable as the resolution increases. In addition, each sequence has a different complexity in terms of the length of the sequence (100–450 frames), the total number of missing pixels (10^5–10^7), the number dynamic occlusions to be solved (1–8), and the label set size per dynamic occlusion (10^6–10^7 offsets). See Table 5.1 for details on each sequence. The input videos and masks, and the inpainting results produced with the proposed method are available in our project page[2]. The first three sequences (*beach-umbrella*, *park-groundtruth*, *park-simple*) are used for comparison with previous methods in Sec. 5.4.2. The remaining three sequences are presented next.

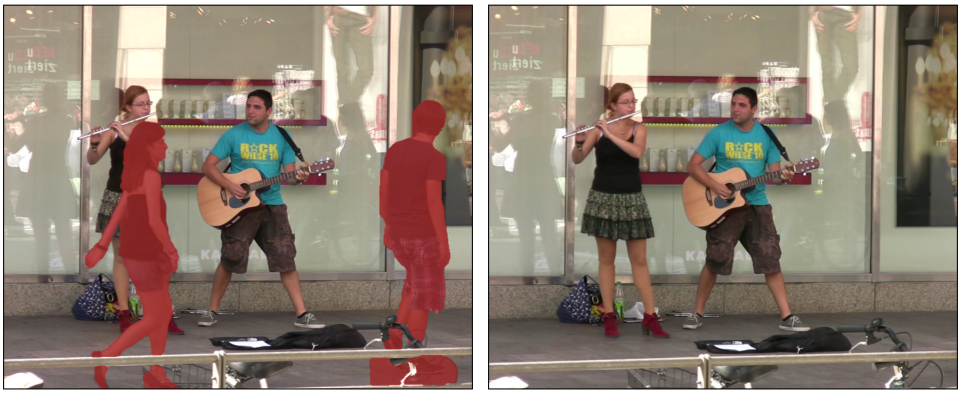
Duo Sequence

In the first high resolution sequence, the *duo* sequence (Fig. 5.5), the task is to remove two pedestrians that occlude two musicians who are standing in front of a reflective surface. These pedestrians occlude the performance of the musicians, which includes repetitive hand movements. In addition, the pedestrians also occlude the reflections of other moving objects in the scene. In particular for this sequence, the dynamic occluded objects to be

[2] `http://www.mpii.de/~granados/projects/vidinp`

Dataset	Video size	Missing pixels	# occluded dynamic objects	Resolution of occluded object	Missing pixels per occluded object	Label set size per occluded object
beach-umbrella	$271 \times 80 \times 98$	10^5	3	32×60	10^5	10^6
duo	$960 \times 720 \times 154$	10^6	2	120×260	10^6	10^6
park-simple	$1440 \times 1080 \times 251$	10^6	1	64×96–80×208	10^4–10^6	10^6
park-complex	$1440 \times 1080 \times 459$	10^6	7	64×44–224×176	10^4–10^6	10^6–10^7
museum	$1440 \times 1080 \times 200$	10^7	8	80×80–384×512	10^5–10^6	10^6–10^7

Table 5.1: Summary of validation video sequences.



(a) Input (b) Inpainted result

Figure 5.5: Inpainting of the *duo* sequence: (a) Overlay between one of the input frames and the mask of the object to be removed; (b) inpainting result using the proposed method.

restored (the musicians), do not change their global positions in space and only undergo localized motions. For this reason, no user-assisted tracking was required for completing this sequence. For the same reason, the set of possible offsets could be restricted to a small range ($[-16, 16]$ pixels) along each spatial dimension, so that the run-time is reduced without excluding any relevant inpainting sources. As shown in Fig. 5.5b, the proposed method provided a plausible completion of the dynamic foreground and background scene elements.

Park-complex Sequence

In the *park-complex* sequence (Fig. 5.6), a person that occludes seven other people is removed. These occlusions are denoted as $o1$–$o7$. The occluded people display different behaviors such as sitting, standing, and walking

Figure 5.6: Completion of the *park-complex* video sequence (top) Crop of the input video frame; (middle) detail of each of the seven occluded dynamic objects $o1$–$o7$ in the video; (bottom) result of the proposed inpainting algorithm, where the person in front is removed.

towards, away from, and parallel to the camera. The scene also contains slight changes of the lighting conditions. The occlusions $o1$, $o6$, and $o7$ correspond to people that are non-periodically moving. In $o7$, the person stands up at the same time that is occluded; in occlusions $o1$ and $o6$, the persons start to walk at the same time that the occlusion occurs. In occlusion $o3$, the person is turning on his vertical axis, and starts to walk away from the camera during the occlusion. His appearance drastically changes after the occlusion occurs as he walks into a shadowed part of the scene. The occlusion $o2$ is particularly challenging: While the body and motion of the person are inpainted plausibly, the proposed method cannot properly inpaint his right arm. Since the person is raising his arm during the occlusion, and this particular type of motion is not available elsewhere in the video, there is no appropriate source for copying the motion. The occlusion $o5$ is another difficult example where the person is occluded by an static object (a bench) at the same time it is occluded by the object marked for removal.

Museum Sequence

The *museum* sequence (Fig. 5.7) is the most challenging dataset. In this sequence, a person that occludes eight other people is removed. Additionally, the people in the scene walk over a specular floor. The occluded people are located at different distances from the camera, and they show different types of motion such as standing, and moving parallel to and away from the camera. The proposed method produced good quality completions for the two high resolution occlusions $o3$ and $o4$. Furthermore, the algorithm successfully completed the reflections on the floor, which is especially noticeable in occlusions $o2$ and $o4$. However, in occlusion $o2$, the person is walking away from the camera. In this case, due to perspective foreshortening, there are no examples in the video that match the exact scale of the occluded object. Despite these challenges, the inpainting method accomplishes a coherent completion with only slight temporal discontinuities. The challenges of inpainting of objects that undergo scale changes are further discussed in Sec. 5.5.

Parameters

The same parameters were used for computing all the results presented in this chapter. These parameters are: $\alpha = \frac{8}{18}$, $\beta = (2\sqrt{2})^{-1}$, $\psi = \frac{1}{2}$, and $\lambda = 0.1$ (see Sec. 5.3.1 for a justification of these specific values). This stability in the parameters shows that our method is robust across different types of scenes.

In addition, the set of possible offsets (or label set) could be further restricted due the specific nature of the test sequences. In these sequences, the occluded objects are always moving over a horizontal ground. This implies that all potential source pixels lie within a narrow range in the y axis. For this reason, the y coordinate of the possible offsets was restricted to $[-16, 16]$ pixels. It is important to note that this constraint was introduced to speed up the experiments, and it does not constitute a requirement of the proposed method.

Lastly, in order to strike a compromise between run-time of the optimization and the quality of the results, the number of cycles of the expansion-move algorithm used to optimize the energy function was restricted to a maximum of five cycles.

Timings

For inpainting each sequence, the run-time of the algorithm fell between 11 hours for the smallest sequence (*beach-umbrella*), and 90 hours for the largest sequence (*museum*). In each sequence, the computation of the individual inpainting sub-problems was done in parallel using a 16-core Xeon X5560 CPU. The running times presented above correspond to the largest

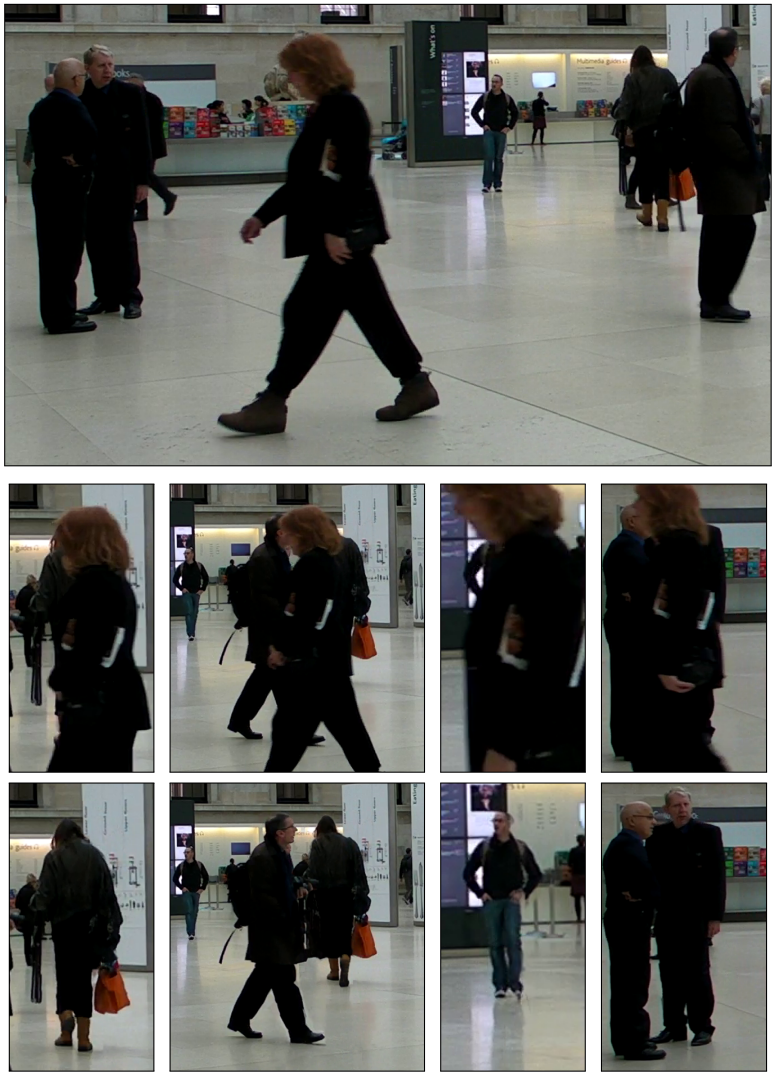


Figure 5.7: Inpainting of the *museum* sequence: (top) Crop of the the input HD video; (middle) detail of the four largest occluded dynamic objects; (bottom) inpainting result, where the woman in the front was removed from the video.

inpainting sub-problem of each sequence. This division of the inpainting problem in several sub-problems is possible since each occlusion can be inpainted independently whenever the occluded objects do not overlap at the moment of the occlusion.

During the preparation of the input, it took around one hour to interactively create the mask of the object to be removed for each video. This was done using the implementation of Video SnapCut [Bai09] available in Adobe After Effects [Adobea]. The user-assisted tracking of the occluded objects for defining each inpainting sub-problem (Sec. 5.3.3) took less than one minute per object.

5.4.2 Comparison to Related Approaches

The proposed energy functional for video inpainting has commonalities with the method of Pritch et al. [Pritch09] for image inpainting, and Wexler et al. [Wexler07] for video inpainting. Nevertheless, the proposed method differs from them in crucial ways. In this section, these differences are discussed, and an empirical comparison of the results is presented.

Relation to the Method of Pritch et al.

The method of Pritch et al. [Pritch09] is designed to work with 2-d images. In this section, a comparison is drawn with a direct extension of their method to video volumes.

The proposed method is similar to Pritch et al. in that it also derives the inpainting result from an offset volume that is estimated using an energy minimization framework based on graph-cuts. Still, there are four major differences between the direct extension of Pritch et al. and the proposed method.

First, the proposed method includes a weighting function γ in the energy (Eq. 5.6) that balances the error distribution along the boundary of the missing region, and inside the missing region. Such a weighting is fundamental for extending Pritch et al. to video sequences as the ratio between pixels at the boundary and inside the missing region is more extreme in videos than in images.

Second, the proposed method introduces a weighting function τ (Eq. 5.8) in order to balance the importance between temporal and spatial inconsistencies in the energy function. This weighting is instrumental in obtaining spatially and temporally coherent inpainting (Sec. 5.4.3).

Third, the proposed method uses a L_2 penalizer (Eq. 5.3), instead of the L_2 square penalizer proposed by Pritch et al. The former penalizer is robuster to outliers caused by larger differences in the appearance of the objects to be completed, and furthermore, it leads to an energy function

(a) Input (b) Wexler et al. (c) Pritch et al. (d) Proposed

Figure 5.8: Comparison with previous inpainting methods. From top to bottom: the *beach-umbrella* sequence (proposed in [Wexler07]), the *park-groundtruth* sequence, and the *park-simple* sequence. (a) Input frame; (b) inpainting result by [Wexler07]; (c) result by an extension of [Pritch09] to videos; (d) result of the proposed method.

that is submodular (see Sec. 2.1.4) so it can be optimized efficiently using graph cuts.

Last, the proposed method computes offsets only for the pixels in the missing region, while the method of Pritch et al. assigns an offset to every pixel in the domain. As a result, their method can potentially change the appearance of the input outside the missing region. This gives their algorithm additional flexibility for computing more plausible inpaintings, at the expense of breaking an essential inpainting assumption, i.e., that only the missing pixels should be modified.

Relation to the Method of Wexler et al.

The main similarity between the proposed method and the method of Wexler et al. [Wexler07] lies in the fact that both methods rely on minimizing an energy function for obtaining the inpainting result. However, the strategy used by both methods is fundamentally different. The proposed method estimates the inpainting as a coherent offset volume, while their method estimates it as the average vote of a set of volumetric patches whose appearance is similar to the context of the missing pixels. This average can lead to blurring artifacts in the inpainting results, or, if a mode estimate is used instead, the results can easily converge to undesired modes. Since the proposed method is not based on a voting scheme, it does not suffer from this type of artifacts.

Additionally, their method uses a local optimizer that does not have guaranties on the quality of the minima obtained, while the proposed method uses graph-cuts to obtain optima whose energy are within a guaranteed bound of the optimum energy.

Lastly, the γ weighting defined in Eq. 5.6 can be seen as a generalization of the weighting proposed by Wexler et al. In their formulation, they assign weights by assuming that the set of missing pixels Ω can be approximated by a spherical region in the space-time volume. In contrast, the proposed weighting removes this assumption and assigns weights according to the actual shape of the missing region.

Empirical Comparison

This section demonstrates how the aforementioned differences with previous methods lead to significant improvements in the results produced by our method. The comparison is performed using in-house implementations of the method of Wexler et al. [Wexler07], and an extension of the 2-dimensional approach of Pritch et al. [Pritch09] to 3-dimensions. The inpainting performance is compared using the video sequences *beach-umbrella*, *park-groundtruth*, and *park-simple* (see Fig. 5.8). In the first sequence, *beach-umbrella*, an umbrella that occludes three walking people is removed.

In *park-groundtruth*, a simulated pedestrian occluder is removed. In *park-simple*, a pedestrian that occludes one another person is removed.

On the low resolution *beach-umbrella* sequence (Fig. 5.8, top row), all the three methods produced plausible results. On *park-groundtruth*, the result of Wexler et al.'s method (Fig. 5.8b, middle row) is overall satisfactory but introduces some inconsistencies on thin structures: A leg and a part of the arm were inpainted with background. The result of an straightforward extension of Pritch et al.'s method to 3-dimensions (Fig. 5.8c, middle row) created a temporally inconsistent inpainting. This could be explained by the absence of the weighting function γ, which makes the cost of introducing such discontinuities at the temporal boundary of the missing region low in comparison to producing a consistent background inpainting inside the hole. This demonstrates the importance of applying the weighting γ in the proposed energy functional. On the other hand, the result of the proposed method plausibly reconstructed the motion of the occluded person (Fig. 5.8d, middle row).

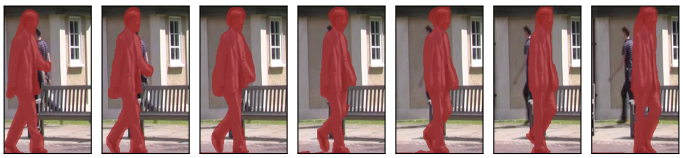
In the *park-simple* sequence (Fig. 5.8, bottom row), the result by Pritch et al. shows a wrong transition to the background behind the person to be removed. The result from Wexler et al. shows a missing thin structure (an arm). In comparison, the result of the proposed method is more plausible, although in some frames the hand was not properly completed. However, this artifact is much less evident than those present in the other two results. Please refer to the supplemental video of this work[3] to fully appreciate the differences described in this section.

5.4.3 Design Validation

The distinctive effect and importance of components of the proposed energy function is demonstrated in this section. These components are: The distance-to-hole-boundary weighting function γ, the temporal weighting τ, and the type of penalizer ψ. Fig. 5.9 illustrates that the inclusion of these two weightings and the use of an adequate penalizer are fundamental for obtaining plausible inpainting results.

Regardless of the penalizer used, if no weighting is performed, the inpainting result corresponds to a background inpainting that disregards the occluded dynamic object, regardless of the type of penalizer applied (Fig. 5.9b, 5.9c). This occurs as the cost of inpainting the background using a single source is lower than the cost of finding several distinct sources for inpainting the dynamic object. On the other hand, if the distance weighting γ is applied but the importance of temporal mismatches is not balanced by the weighing τ, the resulting inpainting is spatially consistent but temporally discontinuous. This effect is illustrated in Fig. 5.9d, where the head is

[3] `http://www.mpi-inf.mpg.de/~granados/projects/vidinp/`



(a) Overlay of input video and occluder mask

(b) No γ weighting, no τ weighting, $\psi = 2$ penalizer

(c) No γ weighting, no τ weighting, $\psi = 1$ penalizer

(d) With γ weighting, no τ weighting, $\psi = 1$ penalizer

(e) With γ weighting, with τ weighting, $\psi = 1$ penalizer (proposed)

(f) With γ weighting, with τ weighting, $\psi = 2$ penalizer

Figure 5.9: Design validation of the proposed energy function: (a) Input video and mask of the object to be removed. (b) Straightforward extension of [Pritch09] to videos. (c) Without the appropriate weighting function, plausible inpaintings cannot be achieved. (d) When using distance weighting γ but no temporal weighting τ, the result is spatially consistent but shows temporal misalignments. (e) The balance between spatial and temporal consistency is kept in the proposed method. (f) Same as configuration as (e) but using a quadratic penalizer (as in (b)). The video corresponds to the occlusion $o5$ of the sequence *park-complex*.

shifted upward, and part of the arm is missing in some frames.

In contrast, the proposed energy function applies the necessary weightings to achieve a plausible result (Fig. 5.9e). Lastly, even when the appropriate weightings are applied, if a penalizer different from ours is used (i.e., a L_2 square penalizer [Pritch09]), the minimization process produces results that suffer from over-smoothing artifacts (Fig. 5.9f).

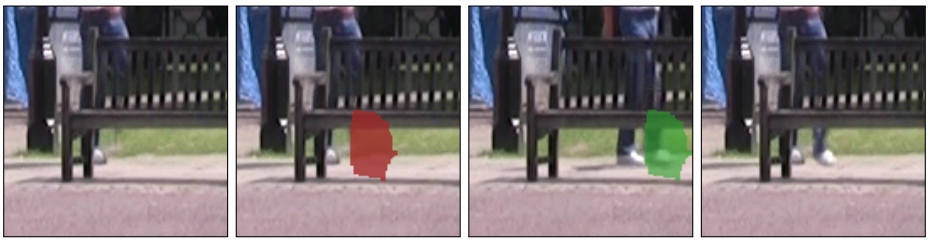
5.4.4 User-Guided Refinement

Although the proposed algorithm can produce more plausible inpaintings than existing approaches (Sec. 5.4.2), it is nevertheless possible that the method fails to produce semantically correct results in some sequences. In many cases, such semantic errors materialize in a contiguous spatio-temporal region of the video volume. This situation is illustrated in Fig. 5.10a, where a leg is missing in the inpainting result.

For addressing this problem, I developed a tool where the user can provide clues to the inpainting algorithm regarding the appropriate source for inpainting a problematic region. To provide such an input, only a few seconds of user interaction are needed. The interaction begins by marking the spatio-temporal region that the user wants to refine. Then, the user selects a suitable source location by selecting the frame number and image region where it is located. At every point of the interaction, the user sees a preview showing a preliminary of the video inpainting result assuming the currently selected source location. After the user is satisfied with the preview, the optimization procedure is rerun using only those sources selected by the user.

Whenever the user constrains the search space to the relevant spatio-temporal region where a suitable source is available, the inpainting algorithm is less likely to select incorrect (but lower energy) sources to complete the missing region. This is illustrated in Fig. 5.10b and 5.10c, where the region with the missing leg, and a proper source region for it are marked by user, respectively. The corrected inpainting is shown Fig. 5.10d. The run-time of this correction step can be much lower than the initial inpainting estimates (usually within minutes) since the missing region and the label set are largely reduced.

This type of user-guided refinement has been previously demonstrated for images [Pritch11, Barnes09], but this work extends it in order to allow the selection of potential source regions that are located at different frames of the video. Please note that all the results presented in this chapter (except Fig. 5.10) were produced without this user-assisted refinement step.



(a) Input (b) Target mask (c) Source mask (d) Refined

Figure 5.10: User-assisted inpainting refinement: (a) Automatic inpainting result, where the leg of the person was incorrectly completed; (b-c) the user marks the target region to be refined (red), and marks a suitable source region in the video volume (green); (d) after computing an inpainting using the constrained source, the error is corrected.

5.5 Limitations

This section discusses the limitations of the proposed method. These can be categorized into limitations due to scene assumptions (motion redundancy, constant illumination), limitations due to simplifications in the design of the energy function (no perspective and scale normalization, speed and acceleration obliviousness), and limitations due to scalability constraints (size of the input videos and missing regions).

Motion Redundancy Assumption

By design, the proposed algorithm determines the color of the missing region by copying the color of other compatible regions in the input video. This is done under the assumption that the appearance of the scene behind the object to be removed is partially available elsewhere in the video. Note that this does not imply that exact instances of the pose of the occluded scene need to visible in other frames for the algorithm to succeed (as in object-based methods). Rather, this assumption implies that there is sufficient redundancy in the video such that the occluded pose can be reconstructed by building a composite of the patches already available in the video. Accordingly, the proposed algorithm will fail if this assumption is not held, for instance, when at the time of occlusion the object to be inpainted had a unique appearance or behavior that is not seen anywhere else in the video volume.

Constant Illumination Assumption

The proposed method is not robust to drastic illumination changes in the scene. This implies that if an occluded object is visible under different il-

lumination conditions in other regions of the video volume, these regions cannot be used as sources for inpainting it. To extend the proposed approach to sequences with illumination variations, an alternative is to perform inpainting in the gradient domain, and solve a 3-dimensional Poisson equation [Roberts99] to construct the final video. A similar approach is applied in Chapter 6 to handle illumination differences arising from viewpoint changes in the video.

Static Camera Assumption

When the camera is static, the appearance of any spatio-temporal region in the video has a very coherent appearance (see Fig. 5.4), and therefore, it can be used to complete other missing regions. However, if the camera moves in a non-linear way, this coherence is lost. This implies that, in order to produce plausible results, the surrogate regions for inpainting any given occlusion have to match not only the appearance of the missing region but also its corresponding camera motion. For this reason, the proposed method cannot be applied to cameras with arbitrary motion. To address this limitation, in Chapter 6 a method is proposed for performing video inpainting in videos taken with moving cameras.

Scale Dependency

The proposed energy function (Eq. 5.2) is not scale invariant, i.e., it cannot complete the appearance of an occluded object using other views of the same object that are in a different scale. Therefore, the proposed method is not expected to produce plausible inpaintings of objects that undergo scale changes due to perspective foreshortening, unless there are instances where the object is visible at the same scale. For instance, this limitation affects objects that move away from or toward the camera. This limitation could be overcome by computing offsets in a scale-space representation of the video, at the expense of enlarging significantly the label set.

Nevertheless, the proposed method can produce plausible reconstructions in such situations provided that the scale does not change significantly during the occlusion. This is demonstrated in the test sequence *museum* in occlusions $o2$ and $o4$ (Fig. 5.7), where the occluded objects are slowly moving away and toward the camera, respectively.

Matching of Speed and Acceleration

The proposed method tests the consistency of adjacent sources by computing their color and gradient distance. However, no tests are performed regarding the consistency of the speed or acceleration of the occluded objects, so the method cannot explicitly encourage preservation of these properties. To

address this limitation, the energy function could be extended to enforce consistencies at higher order derivatives along the temporal axis.

Input Size and Run-time Limitations

The running time of the proposed algorithm can be long, especially on high-resolution sequences (e.g. 90 hours for 200 frames of Full HD video). Nevertheless, the algorithm produces high quality results, and the remaining artifacts can be efficiently remedied with an user-guided interface. This could be considered a step ahead of the industry practice of performing video inpainting manually in a frame-by-frame basis. Still, to speed up the proposed algorithm, fast local solvers such as PatchMatch [Barnes09] could be used. These methods can provide results at interactive rates, as it has been demonstrated for image editing tasks. However, this type of solvers might come at the expense of sacrificing plausibility in the inpainting results.

5.6 Conclusion

This chapter presented a method for removing objects from videos taken with static cameras. The removed objects might occlude not only static objects but also dynamic ones. The resulting method is the first in the literature to demonstrate its applicability to high resolution videos of real scenes featuring several occluded dynamic objects.

The completion of the occluded objects is performed by estimating a offset-volume, where every occluded pixel is completed using the color of an unoccluded pixel whose location is determined by an offset, such that the result looks plausible with respect to the surrounding content. This constraint is encoded using a global energy functional whose minima are designed to correspond to a plausible video where the selected objects are removed.

The resulting optimization problem has a high run-time complexity as the number of offsets is proportional to the number of pixels in the video volume, which is large for high-resolution videos. The run-time is sped up by taking advantage of user-interaction: If the user marks the trajectory of the occluded objects, the space of suitable source pixels can be reduced from the entire video volume to a window around each occluded object. In addition, a similar interaction strategy can be applied in a post-processing step for refining the automatic object removal results, if necessary.

The proposed method was experimentally evaluated using several real-world sequences. These sequences contain diverse motions of a complexity that has not been previously demonstrated in the literature. The results demonstrate that the proposed method produces more plausible video completions than the competing state-of-the-art methods.

CHAPTER 6

Inpainting Static Objects in Moving Cameras

6.1 Introduction

In Chapter 5, an algorithm was presented for inpainting videos taken with static cameras. That algorithm could remove static and dynamic objects from videos (i.e., inpaint them) by reconstructing the appearance of the portion of the video that they occupy. This reconstruction could be done by copying other instances of the occluded objects (or parts of them), assuming that such instances are available on other frames in the video. Other instances are very likely to be available whenever there are no perspective distortions and the motion of the objects in the scene contains some degree of redundancy.

In the case of videos taken with moving cameras, the missing appearance can still be inpainted using the same strategy (i.e., using instances of the occluded objects found on other frames), provided that perspective distortions occurring between frames are properly corrected. Such perspective distortions are extremely challenging to correct for dynamic objects since more detailed scene models are required for modeling the motion of the objects, in addition to the camera motion. On the other hand, perspective corrections can be performed for static objects using only very simple models of the scene geometry (see Sec. 6.3.1). Using such simplified scene models, this chapter presents the first method in the literature to inpaint static scene content in videos filmed using moving cameras under arbitrary motion (i.e.

rotation, translation, and focal length variations). This method has been published in [Granados12b].

The capability of correcting perspective distortions is fundamental for performing plausible video inpainting with moving cameras. There existing approaches to solve this problem assume that a 3-dimensional representation of the scene geometry and the camera projection matrices of every frame are available or can be accurately estimated [Shum00, Bhat07]. However, scanning or manually constructing 3-dimensional models for arbitrary scenes [Debevec98b] is costly and time consuming, and methods that estimate projection matrices and dense depth maps from images or videos are not applicable to general camera motion [Torr99, Pollefeys02b]. The method proposed in this chapter also takes advantage of the geometrical properties of the scene. However, it uses weaker scene models that are flexible enough to handle general camera motion, and therefore, it bypasses the need for performing estimations of camera projection and depth for every frame.

In the next section, the previous video inpainting approaches and the comparative advantages of the proposed method are discussed.

6.2 Previous Work

In Chapter 5, the existing strategies for inpainting videos taken with static cameras were discussed. Most strategies have in common that they synthesize the color of the missing pixels as a combination of video *patches*, i.e., small contiguous 3-dimensional regions in the video volume, which are sampled from other unoccluded parts of the video according to their similarity to the context of the missing pixel. However, the appearance of such patches can vary significantly with camera motion. This variation makes patch comparison difficult, even when the perspective distortions induced by camera motion are not severe. For this reason, these patch-based methods do not generalize well to free-moving cameras (Sec. 6.4).

In order to perform video inpainting with moving cameras, the perspective distortions induced by camera motion need to be corrected. According to this requirement, the existing methods can be classified into two categories: Methods that handle restricted camera motion, and methods that handle general camera motion. These two categories are described next. Other methods that apply an image inpainting algorithm to each frame independently (e.g. [Bertalmío01]) are not discussed since they do not produce, in general, temporally plausible results.

6.2.1 Methods for Restricted Camera Motion

In this category of methods, the strategy is to correct perspective distortions that are well modeled by 2-dimensional homographies. This class of motions

includes panning (horizontal rotation about the center of projection), tilting (vertical rotation about the center of projection), and zooming (variation of the focal length). Cameras that are constraint to these type of motions are called *pan-tilt-zoom* cameras, or PTZ for short, and are commonly found in surveillance applications.

The method of Jia et al. [Jia06] can handle PTZ camera motion, and additionally, camera translation that does not induce severe parallax. First, the static background is inpainted by aligning and copying the color from a background model. This model is constructed as a set of mosaic images, with one mosaic for each depth layer of the scene. These depth layers are defined with the help of the user. Each mosaic is constructed by stitching the corresponding frames into a single panoramic image. The background is inpainted by aligning the reference mosaics to each frame using a homography. Second, the moving foreground objects are detected by background subtraction, and the missing parts are inpainted using a method that assumes periodic motion.

The method of Shen et al. [Shen06] handles PTZ camera motion that can be compensated using a single homography. The missing background and foreground objects are inpainted using a patch-sampling method similar to [Efros99, Wexler07]. In addition, their method can handle the inpainting of moving objects undergoing scale changes by rectifying the perspective distortion with the help of user interaction.

Patwardhan et al. [Patwardhan07] propose an inpainting method where the camera motion is restricted to be parallel to the image plane, in a way that frames can be aligned using only a translation. Similarly to Jia et al., camera motion is handled by constructing a mosaic image of each motion layer. Two motion layers are assumed, one for the static background, and the other for the moving objects. These motion layers are estimated by thresholding the aligned frames. The alignment is done using block-based optical flow. For inpainting both layers, they propose a local method that assigns a priority to every missing pixel; this priority is given based on the number of missing pixels in the neighborhood, and in the presence and direction of edges that might need continuation. Proceeding by highest priority, their method copies those patches that best match the context of the missing pixel.

The algorithm of Venkatesh et al. [Venkatesh09] tracks and segments an occluded object, and constructs a database of segmented frames where it is fully visible. The occluded objects are filled by aligning the corresponding frames in the database to the partially or fully occluded frames in the hole. This alignment is done using dynamic programming. Similar to [Patwardhan07], their method can handle camera motion that is parallel to the camera plane. The camera motion is estimated using block matching, and using this information, a reference background panorama is constructed. This idea was further extended in [Ling09].

The method proposed in this chapter has an important commonality with the method in [Jia06], i.e., a set of homographies are used to align image regions with different depth. In their method, homographies correspond to depth layers, whereas in the proposed method, they correspond to piece-wise planar geometry in the scene. However, depth layers are a less general model for the scene's geometry than piece-wise planar geometry, as the former are often assumed to be parallel to the camera plane while the latter can be arbitrarily oriented. Furthermore, the proposed alignment method does not require user interaction to estimate the geometry (depth layers). Also, it does not constrain the type of camera motion, and therefore, it can handle camera displacements that induce significant parallax.

6.2.2 Methods for General Camera Motion

The most relevant method for video inpainting with free-moving cameras was proposed by Bhat et al. [Bhat07]. In their work, they provide a framework for performing several video editing operations on videos of static scenes taken with moving cameras. These operations include dynamic range and spatial resolution enhancement, object touch-up and replacement, and object removal.

Their method can be divided into two stages. First, they use structure from motion (SfM) for estimating the camera intrinsic and extrinsic parameters, and multi-view stereo (MVS) for obtaining a dense depth map for every frame in the video. In the second stage, these camera parameters and depth maps are exploited for rendering other frames from the view point of the frame to be filled, and using the resulting aligned views to construct a video with the desired editings.

For performing object removal, they propose a video inpainting method that reconstructs the color (and depth) of the missing region by selecting suitable pixels colors from other source frames that are rendered from the viewpoint of the target frame. The selected colors are computed as a composite of coherent patches taken from aligned source frames, following an MRF framework. During the composite computation, the color and depth estimates of the aligned images are used as a guide for selecting compatible sources, in a way that sources with similar depth and color are preferred. Since each frame is reconstructed independently, temporal incoherencies might appear in the reconstruction. They attempt to remove such inconsistencies using a spatio-temporal gradient-domain fusion algorithm.

The main limitation of this approach is the complexity of the input it requires: A reconstruction of the depth and camera location for every frame. Although, structure from motion and multi-view stereo methods can be used for this purpose, such methods do not always succeed in recovering an accurate model for every type of scene.

6.2.3 Relation of the Proposed Method with Previous Methods

The method proposed in this chapter bears some similarities but has fundamental differences to the method of Bhat et al. Both methods inpaint the occluded background by constructing a composite of other aligned source frames where the missing background was visible, and they attenuate the effect of temporal incoherencies using gradient-domain fusion. However, the proposed method is different from Bhat et al.'s in a fundamental way. Our method *does not* require having an estimate of the camera projection matrix and scene depth for every frame in the video. In contrast, such estimates are required by Bhat et al.'s method, and they are obtained using structure-from-motion (SfM) and multi-view-stereo (MVS) methods. This property of not requiring projection matrices and depth estimates is highly desirable since SfM and MVS methods are error-prone and they cannot be applied to every scene and camera configuration (see Sec. 6.4). Instead of re-rendering other frames from the current view using the projection matrices and depth maps (as done in Bhat et al.), the proposed method aligns pairs of source and target frames independently using multiple homographies, and selects the most suitable colors among the aligned sources. This strategy allows the inclusion of just enough geometrical information about the scene in order to handling general camera motion, but without requiring a complete camera calibration and depth estimation for the input video frames.

6.3 Video Inpainting Method

The proposed method consists of three steps. First, pairs of frames are aligned using a set of homographies in a way that perspective distortions are corrected (Sec. 6.3.1). Second, the aligned sources are used as candidates for inpainting the missing region on each frame (Sec. 6.3.2). And third, since the selected sources might have illumination differences, these differences are attenuated using gradient domain fusion (Sec. 6.3.3).

Our method is based on three underlying assumptions. First, it assumes that the region to be inpainted corresponds to static background. Note that this assumption does not imply that the appearance of the background is static, since changes in both camera viewpoint and scene illumination may cause visible color discrepancies between frames. Furthermore, it does not imply that the whole scene has to be static, but only the occluded part has to be inpainted. Second, the method assumes that the missing background is visible in at least one other frame. This assumption is satisfied if the object to be removed or if the camera moves so the background behind is revealed. And third, the method assumes that the scene geometry can be approximated locally using piece-wise planar geometry, in a way that

the objects in the scene visible on different frames can be aligned using homographies. This assumption might not hold for every object in every scene (e.g. for spherical objects or for natural objects such as leafy trees), but nevertheless, high quality inpaintings can still be produced for non-trivial real-world sequences (see Sec. 6.4).

The input to the inpainting method is a video sequence, and a mask for the missing region to be filled (i.e., a mask that marks the object to be removed). Optionally, the user can provide a mask for other dynamic objects that should not be used as sources during the inpainting. The input video is represented as a 3-dimensional volume $\mathbf{V} : \mathcal{I} \otimes \{1, \ldots, T\} \mapsto [0, v_{\max}]^3$, where $\mathcal{I}$ is the set of all pixels in a frame (i.e., $\mathcal{I} = \{1, \ldots, m\} \otimes \{1, \ldots, n\}$ with m and n being the height and width of a frame), T is the number of frames in the video, and $v_{\max}$ corresponds to the maximum output value per color channel of the video camera (e.g. 255 for 8-bit cameras).

The set of missing pixels Ω, and the optional mask for other dynamic objects $\mathcal{F}$, are represented as an index set on $\mathbf{V}$. The t-th frame in $\mathbf{V}$ is denoted as $\mathbf{V}_t$, and the corresponding missing region in the same frame is denoted as Ω_t. The color at pixel $p \in \mathcal{V}$ in the video is denoted as $\mathbf{V}(p)$.

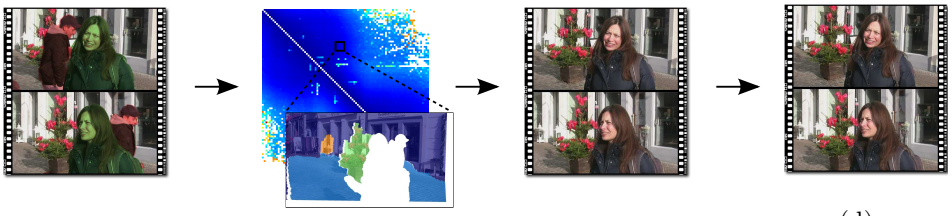
The problem of inpainting the missing region in a *target* frame can be defined as identifying a set of potential *source* frames where the occluded background is visible, and using them to fill the missing region in a plausible way. This definition does not assume that a single source frame can fill the entire missing region; in general, it is necessary to construct a plausible composite of the available sources.

This problem is addressed in two steps: First, correcting for perspective differences between sources and targets (Sec. 6.3.1), and second, constructing a plausible composite of the aligned sources (Sec. 6.3.2). The solution of each of these tasks is defined as the minimum of a global energy functional, which is obtained using efficient combinatorial optimization based on graph cuts (see Sec. 2.1.3). Additionally, a third post-processing step is applied in order to attenuate potential artifacts caused by differences in illumination between frames (Sec. 6.3.3). The overall pipeline is illustrated in Fig. 6.1.

6.3.1 Frame Alignment

The first task of the algorithm is to correct for the perspective differences between a target frame $\mathbf{V}_t$ with missing regions and any potential source frame $\mathbf{V}_s$. If both frames are aligned, it is possible to inpaint a pixel location p in the target using the color of the aligned source at the same location. The alignment process can be defined as finding a mapping $F_{st} : \mathcal{I} \mapsto \mathcal{I}$ such that each target color $\mathbf{V}_t(p)$ is as similar as possible to its corresponding source color $\mathbf{V}_s(F_{st}(p))$, for every $p \in \mathcal{I}$.

In the computer vision literature, there exist well established methods for estimating the mapping F_{st}. Methods such as optical flow and stereo



(a) Input video and mask

(b) Pair-wise frame alignment

(c) Inpainting

(d) Gradient-domain fusion

Figure 6.1: Proposed inpainting pipeline. (a) The input is a video, a mask for the object to be removed (shaded in red), and a mask of other dynamic objects in the scene (shaded in green). (b) The input frames are pairwise aligned based on a set of local homographies. (c) The inpainting result is composited by minimizing a global energy functional that encodes the difference between the selected source frames and a reference inpainting given by the weighted average of the aligned frames. (d) In a post-processing stage, gradient-domain fusion is performed to remove potential illumination discrepancies.

matching could be applied if the source and target frames did not contain missing regions. However, in the presence of missing regions (as it occurs in our setting), such methods cannot be directly used since the alignment is not properly defined for those regions. For this reason, an alignment algorithm needs to be designed such that it also aligns the missing regions despite the absence of color information inside them.

To address this design requirement, the homography was selected as a basic element for alignment. Homographies are suitable for this task since they can align a missing region based on the geometric matches found in the visible regions around it. This is a good approximation if the geometry of the missing region can be assumed to be locally planar. However, a single homography may not provide a reasonable estimate of F_{st}, since it can only align one of the (possibly many) planar regions in the scene. To overcome this limitation, the proposed method extends this strategy: If the geometry in the scene can be approximated using piece-wise planar geometry, the algorithm aligns two images by decomposing them into regions that are each placed into correspondence using a different homography. For the entire frame, this results in a set of homographies needed to perform the alignment.

Homography-based Alignment

To obtain the alignments F_{st}, the proposed method has two stages: First, it computes a set of candidate homographies between the source and target frames, and second, for every pixel it selects a single homography such that the alignment error is minimized. The result of this alignment process is

(a) Target frame and mask

(b) Source frame

(c) Partial fill using aligned source

(d) Homography map of alignment

Figure 6.2: Homography-based frame alignment for inpainting. (a) Input frame where the region to be inpainted is shaded in red. (b) A source frame where the region to be inpainted is partially visible (the remaining parts would need to come from other sources). (c) The target frame is partially filled using the aligned source. (d) Overlay between the aligned source and the mapping K that selects the homography to be used to align each region. The linearity of homographies allows the algorithm to effectively extrapolate into the region to be inpainted, for which no reference colors are available during the alignment process.

illustrated in Fig. 6.2.

The first stage, finding candidate homographies, starts by establishing geometrically consistent feature correspondences between all pairs of frames [Hartley04]. This is done by finding potential feature point correspondences, and discarding outliers that do not satisfy the epipolar constraint (defined in the next subsection). For consecutive frames pairs, potential feature correspondences are obtained by KLT tracking [Lucas81, Shi94]. For non-consecutive frame pairs, potential correspondences are found by performing approximated nearest neighbor search [Muja09] of SURF features [Bay08]. The epipolar constraint is tested on every pair of frames by thresholding the distance between matched feature points and its corresponding epipolar line. This test requires estimating the fundamental matrix, which is obtained using RANSAC [Fischler81] over the set of matched feature locations for every frame pair [Hartley04].

Once geometrically consistent feature correspondences are obtained between each source and target frame, a set of homographies are incremen-

Data: A pair of frames $\mathbf{V}_s, \mathbf{V}_t$
Result: A set of homographies H_{st} from $\mathbf{V}_s$ to $\mathbf{V}_t$
p_s = feature points in $\mathbf{V}_s$;
p_t = feature points in $\mathbf{V}_t$;
m_{st} = feature matches between p_s and p_t;
$H_{st} = \{\}$;
$i = 1$;
while $|m_{st}| > 6$ *and* $i \leq k_{max}$ **do**
  H_{st}^i = homography estimation from feature matches m_{st};
  m_{st}^{in} = inliers of homography estimation above;
  $m_{st} = m_{st} \setminus m_{st}^{\text{in}}$;
  $i = i + 1$;
end

Algorithm 2: Estimation of candidate homographies between a source and target frame.

tally estimated for each pair: At step 1, a homography is estimated using RANSAC on the whole set of feature correspondences. At step n, the feature correspondences determined as outliers at step $n-1$ are used to estimate a new homography. This process is iterated until either $k_{\max}$-homographies are determined, or there are not enough feature correspondences to continue. This process is illustrated in Algorithm 2.

In the second stage, a homography is selected for mapping each source pixel to the target. The homography selection needs to satisfy two objectives: The resulting alignment error between the source and target needs to be minimized, and the boundary between adjacent regions aligned by different homographies needs to look plausible. This alignment is obtained by minimizing an energy functional, which is described next.

Energy Function for a Homography-based Alignment

The homography selection problem can be defined as follows. Let $H_{st} = \{H_{st}^1, \ldots, H_{st}^k\}$ be the set of candidate homography matrices that align parts of $\mathbf{V}_s$ to $\mathbf{V}_t$. The objective is to compute a map $K : \mathcal{I} \to [1 \ldots k]$ that determines the homography $H_{st}^{K(p)}$ that best aligns each pixel p at the source to the target. This map can be obtained by minimizing the energy functional

$$\mathcal{E}(K) = \sum_{p \in \mathcal{I}} \underbrace{D_p(K(p))}_{\text{data term}} + \beta \sum_{(p,q) \in \mathcal{N}(\mathcal{I})} \underbrace{\mathbb{1}_{\{K(p) \neq K(q)\}} V_{p,q}(K(p), K(q))}_{\text{prior term}}, \quad (6.1)$$

where $\mathcal{N}(\mathcal{I})$ denotes a spatial neighborhood system (4-neighbors in the current algorithm), and the factor β balances the importance of between minimizing the alignment error (the data term D_p) and maximizing the agreement between adjacent pixels that are mapped using different homographies

(the prior term $V_{p,q}$). The factor β determines the granularity of the segmentation of each frame pair into regions mapped by different homographies: Lower values of β imply that smaller contiguous frame regions will be mapped by the same homography, whereas larger β implies that larger regions will be mapped using the same homography, leading to overall fewer homographies being used for aligning the frame pair. This factor was set to $\beta = 10$ upon manual inspection of the results in a reference sequence (Fig. 6.2), and it was left constant for all the experiments presented in this chapter.

Let p_h denote the pixel location $p = (x, y)$ expressed in homogeneous coordinates, i.e., $p_h = (x, y, 1)$. If K^* is the labeling that minimizes Eq. 6.1, the alignment F_{st} is given by

$$F_{st}(p) = H_{st}^{K^*(p)} p_h, \tag{6.2}$$

so that the source color corresponding to a given missing pixel $\mathbf{V}_t(p)$ is given by

$$\mathbf{V}_s(F_{st}(p)) = \mathbf{V}_s(H_{st}^{K^*(p)} p_h). \tag{6.3}$$

The prior term $V_{p,q}$ penalizes the pairwise color discrepancies between two adjacent pixels p, q in the source frame when they are aligned using distinct homographies H_{st}^u, H_{st}^v. This discrepancy is computed as

$$V_{p,q}(u, v) = \|\mathbf{V}_s(H_{st}^u p_h) - \mathbf{V}_s(H_{st}^v p_h)\|_2 + \|\mathbf{V}_s(H_{st}^u q_h) - \mathbf{V}_s(H_{st}^v q_h)\|_2 . \tag{6.4}$$

The data term D_p measures the color differences between the source and target frames if aligned using the homography H_{st}^k. First, in cases where both the source and the target pixels are both not in a hole region, this difference is computed as

$$d_p(k) = C_{st}^k(p)||\mathbf{V}_t(p) - \mathbf{V}_s(H_{st}^k p_h)||_2, \tag{6.5}$$

where C_{st}^k is a compatibility weight, defined in the next section. Second, whenever the source pixel is missing, the mapping is assigned an infinite (or a very high) cost. This discourages selecting a homography H_{st}^k if it produces a mapping to a location in the source that is marked as missing. And third, all homographies that map missing target pixels to valid locations in the source are given a zero (or very low) cost. This implies that any homography can be used for inpainting the missing region regardless of its color, and consequently, that the homography selected for a given missing pixel will only depend on the alignment error of the (non-missing) pixels around it. In this way, the alignment error available at the boundary of the missing region is *propagated* (via the prior term) to the interior of the missing region. The final data term is defined as

$$D_p(k) = \begin{cases} \infty & \text{if the source } \mathbf{V}_s(H_{st}^k p_h) \text{ is a missing color,} \\ 0 & \text{if the target } \mathbf{V}_t(p) \text{ is a missing color,} \\ d_p(k) & \text{otherwise.} \end{cases} \tag{6.6}$$

The proposed energy functional (Eq. 6.1) is minimized using the expansion move algorithm (see Sec. 2.1.3), where the label set corresponds to the number of candidate homographies estimated between a single source and target pair. This energy functional is minimized independently for every frame pair in the video sequence. Since each minimization problem is independent, the alignment process can be easily parallelized.

Compatibility Between Homography and Epipolar Geometry

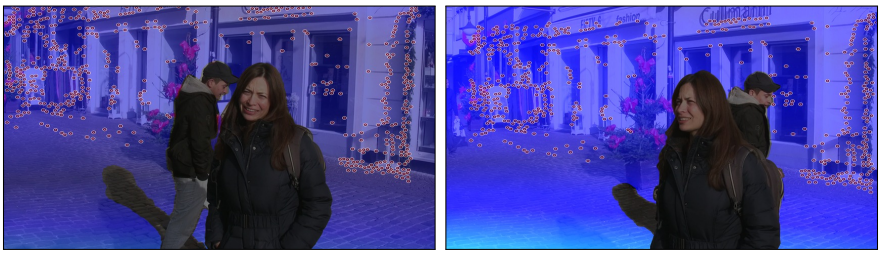
For a pair of frames s, t, the *epipolar constraint* is determined by the *fundamental matrix* f_{st} as follows: If two points p_s, p_t on frame s, t respectively correspond to the projection of the same 3D point in the scene, then the points p_s, p_t satisfy the epipolar constraint $p_t f_{st} p_s = 0$ [Hartley04]. Please recall that for a given frame pair s, t, the epipolar line l_t corresponding to a point p_t is defined as the projection on frame s of the ray passing through p_t and the camera center of frame t. Therefore, the fundamental matrix encodes the epipolar line $l_t = p_t f_{st}$ in frame s corresponding to the point (seen as a ray) p_t in frame t. If p_s and p_t are projections of the same 3D point, then p_s lies on the epipolar line l_t and therefore, $l_t p_s = (p_t f_{st}) p_s = 0$.

The epipolar constraint is exploited for testing the *compatibility* at pixel p between the candidate homography H^k_{st} and the epipolar constraint of the frame pair (Fig. 6.3). In Eq. 6.5, the compatibility factor $C^k_{st}(p)$ aims at encoding the following observation: A single homography is very unlikely to provide a good alignment for the whole frame (unless the scene is actually composed of a single plane). Therefore, each homography should only be used to align those image regions where it fits well the geometry of the scene. Since the geometry is unknown, this test is approximated using the epipolar constraint between the two frames. Following this criterion, the compatibility factor should decrease with the distance $(p_h f_{st})(H^k_{st} p_h)$ between the epipolar line of p in frame t (i.e., $p_h f_{st}$), and its location predicted by the homography (i.e., $H^k_{st} p_h$). Using this criterion, the compatibility factor is defined as

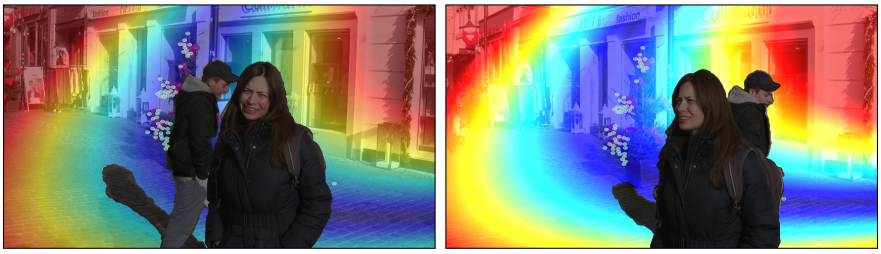
$$C^k_{st}(p) = 1 - \left[\exp\left(-\frac{1}{2} \frac{\left((p_h f_{st})(H^k_{st} p_h) \right)^2}{\mathrm{r}^2} \right) - \frac{1}{2} \right], \tag{6.7}$$

where r corresponds the inlier distance threshold of RANSAC defined during the estimation of the homography and fundamental matrices.

Unlike for methods based on structure from motion [Bhat07], in the proposed method a unique estimate of the fundamental matrix is not critical for computing a correct alignment. In particular, when the camera motion or the scene geometry are not general (e.g. the camera rotates about the center of projection or all feature points in the scene lie on a plane), the fundamental matrix is not unique [Torr98]. This occurs as the resulting set of



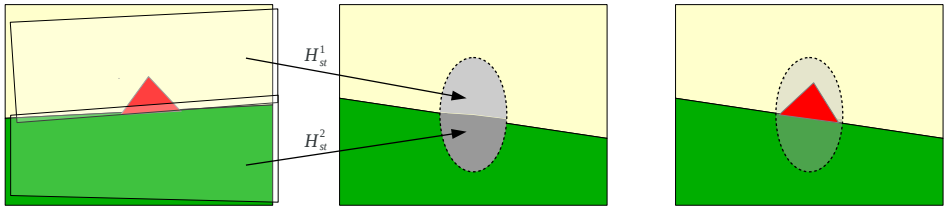
(a) Homography corresponding to the facade



(b) Homography corresponding the tree

Figure 6.3: Spatially varying compatibility between homographies and the epipolar constraint (a) Compatibility map for the homography that aligns the facade of the scene (blue: compatible, red: incompatible). (b) Compatibility map for the homography that aligns the ornament plant in the middle of the scene. These frames correspond to the source and target frames shown in Fig. 6.2. The feature points used for estimated the homographies are shown on each frame pair. Note that the regions containing key points have a higher compatibility score. This compatibility map allows determining the regions in the frame are suitable to be aligned using a given homography, even for regions where no feature matches were found.

feature correspondences cannot provide enough constraints to determine the fundamental matrix uniquely (which has eight degrees of freedom). In this case, the result is a class of fundamental matrices that satisfy the epipolar constraints, but where only one of them represents properly the geometry of the scene. Nevertheless, the proposed method uses the estimated fundamental matrix *only* for calculating the compatibility weight in Eq. 6.7. These weights become uniform when the fundamental matrix is degenerate, as all feature points are equidistant to the corresponding epipolar line. Therefore, the proposed algorithm is undisturbed by degenerate cases, but it still can take advantage of fundamental matrices whenever they are available. For this reason, the proposed method can be used even when the input scene does not allow the estimation of a unique fundamental matrix (e.g. camera rotation about its center or projection, or scenes containing a single plane).



(a) Source (b) Target (c) Inpainted target

Figure 6.4: Illustration of the frame alignment process. The source frame (a) is aligned to the target frame (b) using a set of homographies (H^i_{st}). This is done in order to provide inpainting candidates for the missing region at the target (shown as a gray area in (b)). The proposed algorithm selects, for each pixel in the source frame, a homography to be used for aligning it to the target. As a result, different regions in the missing region be filled using source regions aligned via different homographies (the selected homographies are denoted by shades of gray in (b)). The missing region is filled by a simple copy of the aligned sources (c).

Avoidance of Repetition Artifacts

In the alignment process defined so far, it is possible that a source pixel be mapped to multiple pixels in the target. Formally, this occurs if two different pixels p, p' in the target are mapped to the same source pixel q, via different homographies with index k, k' defined such that $H^k_{st} q_h = p_h$ and $H^{k'}_{st} q_h = p_h$. This situation allows the alignment algorithm to copy multiple times the same source object to the target frame. Although, this repetition is not an issue in regions of uniform appearance, in highly structured regions, it can lead to artifacts by duplicating structures that should be unique. In the proposed algorithm, this situation is prevented by reversing the aligning process, i.e., by aligning the target frame to the source instead of aligning the source to the target. The resulting alignment can be represented as the mapping F_{ts} where $\mathbf{V}_t(F_{ts}(p)) \approx \mathbf{V}_s(p)$, where each target pixel can be assigned to several sources. The final alignment is produced in two steps: First, each source pixel is labeled with only one of the target pixels assigned to it (the choice of which is arbitrary); second, this labeling is projected to the target frame using the inverse homography of the corresponding region. In this way, repetition of source pixels is prevented in the alignment result. The final structure the alignment process is illustrated in Fig. 6.4.

Homography Pruning

In order to keep the label set as small as possible, in a pre-processing step, unsuitable homographies are removed from the candidate set. Unsuitable homographies are defined as transformations that do not satisfy one of the

following three criteria: The homography is orientation preserving; it produces a proportionate scaling along both axis, i.e., the ratio between the first two eigenvalues of the corresponding affinity matrix is not larger than a threshold (set to 0.1); and it produces an area scaling that does not vary too much with position. The latter criterion is introduced to avoid situations where the appearance of the target cannot be properly reconstructed from the source due to discretization. This criterion can be evaluated by ensuring that the norm of the projectivity vector of the homography is not larger than a threshold (set to 0.1). Although the last criterion excludes situations that can be found in practice, such homographies are unlikely to be correctly detected. This occurs as most interest point detectors are only invariant up to affinity transformations, and therefore, they are not invariant to extreme perspective distortions.

6.3.2 Scene Composition

The frame alignment described in the previous section provides a inpainting estimate of the target frame $\mathbf{V}_t$ using parts of a source frame $\mathbf{V}_s$. In general, there exist several source frames that partially or completely cover the missing region Ω_t in frame t. For each pixel in Ω_t, a single source must be selected among the available (aligned) frames in a way that the resulting set of sources corresponds to a plausible inpainting.

Energy Function for Scene Composition

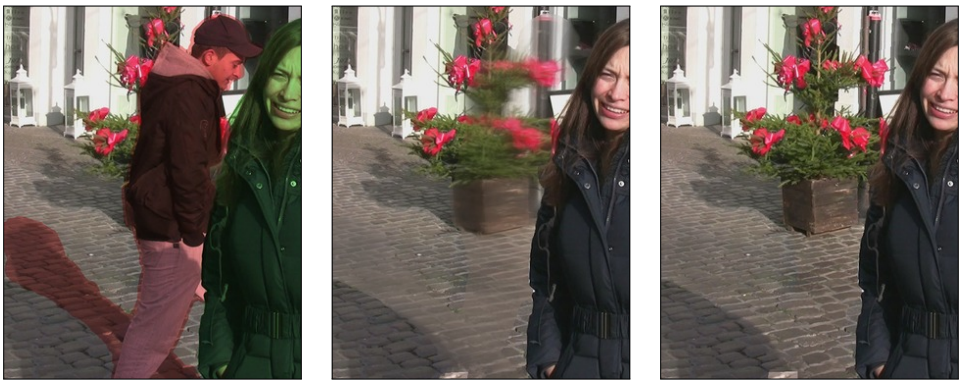
The process of constructing a plausible composite can be formalized as follows. Let $S_t : \Omega_t \mapsto \{1 \dots T\}$ be the mapping specifying a source frame for every missing pixel $p \in \Omega_t$ in frame t. S_t is obtained by minimizing the energy functional

$$\mathcal{E}'(S_t) = \sum_{p \in \Omega_t} \underbrace{D'_p(S_t(p))}_{\text{data term}} + \gamma \sum_{(p,q) \in \mathcal{N}(\Omega_t)} \underbrace{\mathbb{1}_{\{S_t(p) \neq S_t(q)\}} V'_{p,q}(S_t(p), S_t(q))}_{\text{prior term}}, \quad (6.8)$$

where D'_p and $V'_{p,q}$ are the data and prior term, respectively, and γ controls their relative importance (set to $\gamma = 10$). The neighborhood system contains all pairs of adjacent pixels (p, q) where p belongs to the missing region, i.e., $\mathcal{N}(\Omega_t) \equiv \{(p, q) : p \in \Omega_t, q \in \mathcal{I}\}$.

The prior term $V'_{p,q}(u, v)$ should assign a high cost to results that contain adjacent sources that do not look plausible. To approximate this criterion, it is defined such that it measures the color discrepancy between two distinct source frames u, v when they are selected for filling two adjacent missing pixels p, q, respectively. This discrepancy is given by

$$V'_{p,q}(u, v) = \|\mathbf{W}^t_u(p) - \mathbf{W}^t_v(p))\|_2 + \|\mathbf{W}^t_u(q) - \mathbf{W}^t_v(q)\|_2, \quad (6.9)$$



(a) Input and mask (b) Average of sources (c) Inpainting composite

Figure 6.5: The weighted average of aligned source frames is taken as guide for the optimization of the final inpainting composite.

where $\mathbf{W}_u^t(p)$ denotes the source frame $\mathbf{V}_u(F_{ut}(p))$ after being aligned to the current target $\mathbf{V}_t$ (see Eq. 6.3).

On the other hand, the data term D'_p should assign a high cost to sources that do not agree with the true background that is behind the object to be removed. Since this background is unknown, an approximation can be obtained by computing the average color of all sources for a given pixel. This approximation is given by

$$\mathbf{R}_t(p) = \frac{\sum_{l=1}^{T} a_l^t \mathbf{W}_l^t(p)}{\sum_{u=1}^{T} a_l^t}, \tag{6.10}$$

where the color of each pixel p corresponds to the weighted average of the candidates (see Fig. 6.5b). The definition of the weighting a_l^t is provided in the next section.

The data term assigns a high cost to sources $\mathbf{W}_u^t(p)$ that have a large color difference with the reference average $\mathbf{R}_t(p)$. This difference is given by

$$D'_p(u) \quad = \quad \|\mathbf{W}_u^t(p) - \mathbf{R}_t(p)\|_2. \tag{6.11}$$

This term is evaluated only on cases where the candidate color $\mathbf{W}_u^t(p)$ is properly defined, i.e., when the following three criteria are satisfied: A correct alignment was found between the target and source frames; the corresponding source pixel is not a hole (or foreground) pixel; and the projection of the source pixel on the target frame lies within the image domain. Sources that do not satisfy these criteria are excluded from the set of possible solutions. This is done by assigning these configurations an infinite (or very high) cost.

Alignment Score

In Eq. 6.10, the *alignment score* a_l^t represents the confidence of the alignment between frames l and t, which is computed over the mutually visible, unoccluded regions between the two frames. The alignment score a_l^t is defined as

$$a_l^t = \frac{\sum_{p \in \mathcal{I} \setminus \Omega_t} D(p, \partial\Omega_t) \|\mathbf{W}_l^t(p) - \mathbf{V}_t(p)\|_2}{\sum_{p \in \mathcal{I} \setminus \Omega_t} D(p, \partial\Omega_t)}, \tag{6.12}$$

where the distance weight

$$D(p, \partial\Omega_t) = \exp\left(-\frac{\mathrm{d}(p, \partial\Omega_t)}{2\sigma_d}\right) \tag{6.13}$$

is high for misalignments located closer to the boundary of the missing region $\partial\Omega_t$. In the distance weight, $\mathrm{d}(p, \partial\Omega_t)$ corresponds to the distance between a pixel p and the boundary of the hole, and σ_d controls the fall-off of the score, which is set to $\sigma_\mathrm{d} = 8$ pixels.

Candidate Sources

The energy functional $\mathcal{E}'$ (Eq. 6.8) is minimized independently for every frame containing missing regions. For each of those frames, a suitable set of candidate source frames needs to be defined. To define this set of candidates, several strategies could be devised. For instance, a frame alignment could be computed between each target frame and every other frame in the video (i.e., T source frames per target), resulting in a total of T^2 frame-to-frame alignment operations (see Fig. 6.6). However, since the optimization of Eq. 6.8 is linear on the number of candidate source frames, it is advantageous to constraint this set in order to reduce the run-time of the inpainting algorithm. Therefore, an alternative to reduce the run-time is to define a sliding window of n source frames around each target frame, resulting in a total of nT alignment operations. Such a sliding window corresponds to the temporal neighborhood of the target frame, and therefore, it should contain the most similar source frames assuming that the camera motion is smooth. In the proposed method, one of these two strategies (T^2 or nT alignment operations) is adopted depending on the number of frames in the sequence, such that the total number of alignment operations is kept under a desired maximum bound (50^2 alignments in our experiments).

Other sampling strategies could be used to reduce the number of candidate source frames, such as randomized sampling and region growing [Barnes09]. These alternative sampling methods will be investigated in the future.

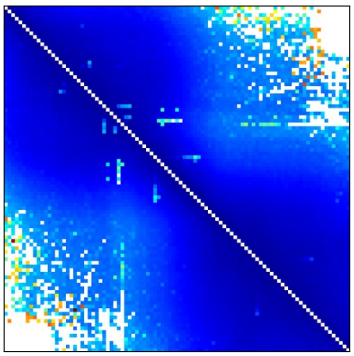


Figure 6.6: Alignment score matrix. Each entry (s, t) represents the average alignment error between frames s and t (blue: low alignment error, red: high alignment error, white: no alignment was found). Large scores close to the diagonal correspond to pairs for which no proper alignment was found.

Minimization of Energy Functional

The energy functional in Eq. 6.8 is minimized using the expansion move algorithm described in Sec. 2.1.3, where the label set corresponds to the number of candidate frames (T or n) in the video sequence. This minimization process is applied independently to every frame in video containing missing regions. Since each minimization problem is independent, this stage of the inpainting algorithm can be easily parallelized.

6.3.3 Handling of Illumination Mismatches

In the input video sequence, it is possible that there exist minor color differences between source and target frames that show the same object. These differences can be caused by changes in illumination (e.g. in outdoor scenes), or by differences in the light reflected by the surfaces (e.g. reflection of lambertian surfaces depends on the direction of the light and the observer).

The proposed energy functional assigns higher cost to sources with inconsistent colors, and therefore, sources with consistent illumination are preferred by the optimization procedure. Despite this fact, when the illumination of target frames is different from all source frames, the proposed algorithm can produce noticeable boundaries (see Fig. 6.7b).

To address this problem, gradient-domain fusion is performed on the inpainting result. The fusion is performed by solving the Poisson equation with Dirichlet boundary conditions. This is a well established strategy in image editing to remove potential illumination differences [Pérez03]. However, if this strategy is applied to every frame independently, it can introduce flickering artifacts. To prevent these artifacts, an additional cost is introduced that is proportional to the color differences between the inpainted

objects and the corresponding objects in the previous (blended, inpainted) frame, after being aligned using optical flow [Sun10].

Formally, let $\{f_p^*\}_{p\in\mathcal{I}}$ be the set of pixel colors of the current (inpainted) frame, and let $\{g_p^*\}_{p\in\mathcal{I}}$ be the set of pixel colors of the previous (inpainted, blended) frame.

The Poisson equation minimizes the differences between the gradients in the input image and the blended image. This difference is given by $d^s(p,q) = |(f_p - f_q) - (f_p^* - f_q^*)|$, using a first order approximation of the gradient, where p, q are adjacent pixels in the image domain. However, the gradient is not uniquely defined for pair of pixels that have different source frames, i.e., each source provides a (potentially) different gradient value. To resolve this issue, the proposed fusion method does not penalize gradient differences at the boundary between source regions (i.e., it assumes a zero gradient at these locations). This formulation leads to the cost function

$$d^s(p,q) = \mathbb{1}_{\{S(p)=S(q)\}} \left|(f_p - f_q) - (f_p^* - f_q^*)\right|, \tag{6.14}$$

where $S(p), S(q)$ correspond to source frame indices selected for inpainting pixels p, q, respectively.

In order to cope with temporal flickering, an additional cost function is introduced:

$$d^t(p) = |f_p - g_p^*|. \tag{6.15}$$

This function assigns a low cost to a blended color f_p if it is close to the color in the previous (aligned) frame.

The Poisson-blended colors $f = \{f_p\}_{p\in\mathcal{I}}$ for the current frame can be obtained by minimizing the energy functional

$$\begin{aligned}\mathcal{E}(f) &= \sum_{(p,q)\in\mathcal{N}(\Omega)} d^s(p,q)^2 &&+\lambda \sum_{p\in\Omega} d^t(p)^2 \\ &= \sum_{(p,q)\in\mathcal{N}(\Omega)} \Big((f_p - f_q) - (f_p^* - f_q^*)\Big)^2 &&+\lambda \sum_{p\in\Omega} (f_p - g_p^*)^2 \end{aligned} \tag{6.16}$$

where Ω is the set of missing pixels in the current frame, and $\mathcal{N}(\Omega)$ denotes the set of adjacent pixels (p, q) where at least p belongs to Ω. The scalar λ is a weight that controls the relative importance of the spatial and temporal cost functions. In order to give equal importance to both, this scalar is set to the ratio between the number of spatial and temporal costs, i.e., $\lambda = \frac{|\mathcal{N}(\Omega)|}{|\Omega|}$. The rationale behind this design is analogous to the temporal weighting α defined in Chapter 5 used in Eq. 5.8.

The blended image f with minimum cost can be obtained by solving the linear system $\frac{\partial \mathcal{E}(f)}{\partial f_p} = 0$, which is given by

$$(|\mathcal{N}_p| + \lambda) f_p - \sum_{q\in\mathcal{N}_p\cap\Omega} f_q = \sum_{q\in\mathcal{N}_p\cap\mathcal{I}\setminus\Omega} f_q^* + \sum_{q\in\mathcal{N}_p} (f_p^* - f_q^*) + \lambda g_p^*, \tag{6.17}$$

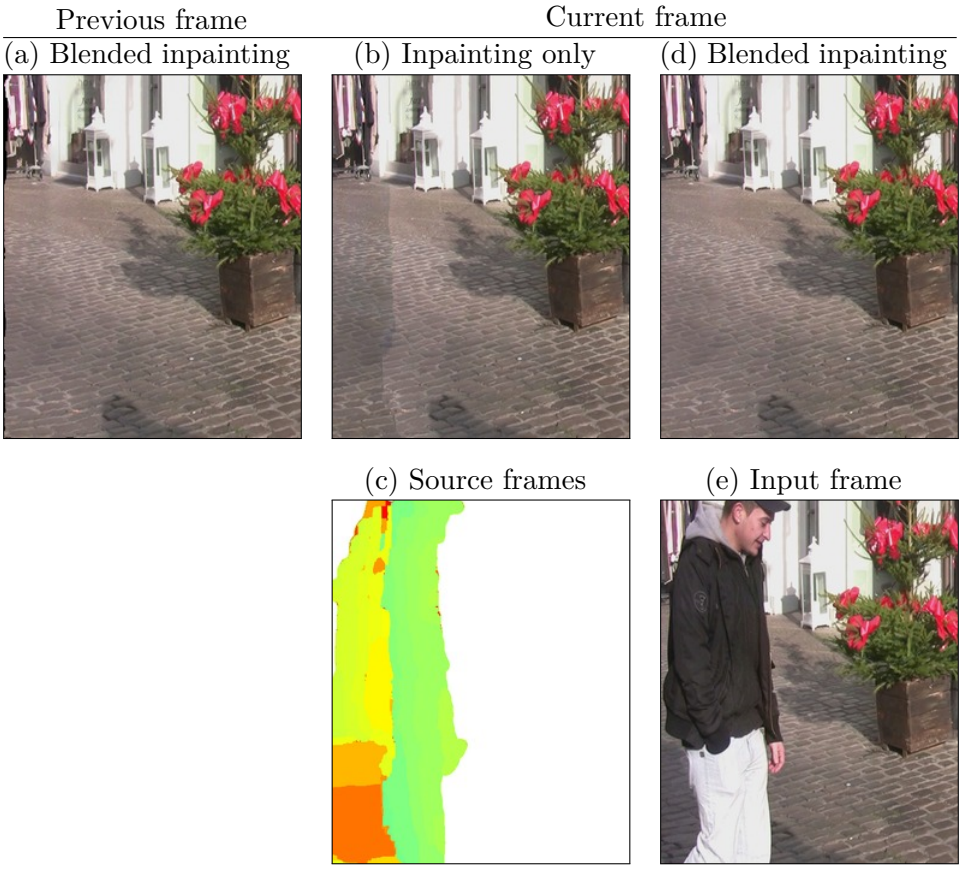


Figure 6.7: Removal of illumination differences using Poisson blending: The blended inpainting of the previous frame (a) is aligned using optical flow to the inpainting of the current frame before blending (b), where illumination differences are visible at the boundary between source regions (c). The source regions in (c) are shown in a color-code corresponding the time-stamp of the source frame. The blended result (d) is computed using gradient-domain fusion. The input frame (e) is shown for comparison.

where $\mathcal{N}_p$ denotes the set of adjacent pixels of the pixel p. This linear system is solved using the conjugate gradients method. The effect of the blending process on the inpainting result is illustrated in Fig. 6.7.

6.3.4 Differences with Depth-based Inpainting Methods

Bhat et al. [Bhat07] provide an alternative to the approach proposed in Sec. 6.3.1 for aligning the source and target frames. Their strategy consist in rendering the source frames from the view point of the target frame. This rendering is possible if the camera projection matrix and the depth of each pixel are known for both the source and target frame. The camera projection matrices are recovered using structure-from motion (SfM) [Snavely06], and depth maps are estimated using a multi-view-stereo (MVS) method. This strategy is also known as image-based rendering (IBR) [Shum00].

However, the necessity of having camera projection estimates introduces two limitations. First, structure from motion methods require that the translation of the camera be sufficiently large as to properly triangulate the location of the set of matching interest points. Second, since SfM is sensitive to the initial estimate of the focal length of the camera (which is usually unknown), it might fail to provide correct projection matrix estimates on sequences where the focal length is variable. These limitations reduce the set of videos where the inpainting method can be applied. In contrast, the proposed algorithm does not suffer from these restrictions on the camera configuration, since an estimation of the camera projection matrix is not required.

On the other hand, multi-view stereo methods require that either the scene be completely static [Bhat07], or that the scene be simultaneously captured from different viewpoints using many cameras. Additionally, MVS methods often assume Lambertian surfaces. In contrast, the proposed algorithm aligns pairs of frames based only on feature point matches, without requiring that the scene be static or the surfaces be lambertian. This is achieved by following a different assumption: The scene can be approximated as a set of planar surfaces. If this assumption is satisfied, the alignment can be performed using a set of homographies.

In summary, the proposed method recovers only as much geometrical information about the scene as required for performing plausible inpaintings (only homographies and fundamental matrices). In this way, the restrictions of SfM and MVS are avoided. In the limit, when arbitrarily many homographies are allowed, the proposed alignment method corresponds to a stereo method capable of handling missing regions.

Sequence	Resolution	Length	Camera baseline	Focal length
S1	1440×1080	95	Narrow	Fixed
S2	1440×1080	100	Narrow	Fixed
S3	960×720	180	Wide	Variable
S4	960×720	270	Narrow	Fixed
S5	960×720	225	Wide	Fixed
S6	960×720	220	Wide	Fixed
S7	1440×1080	80	Narrow	Variable

Table 6.1: Summary of the test sequences used for experimental validation

6.4 Experimental Validation

In this section, the proposed algorithm is validated using seven video sequences taken with a hand held camera, including some with variable focal length. In addition, the advantages of the proposed method are illustrated in relation to methods that do not account for camera motion, and methods based on multi-view stereo.

6.4.1 Experimental Setting

For empirically validating the proposed algorithm, seven real-world sequences were acquired on four different scenes (see Fig. 6.8). These sequences are named S1–S7, and they are available on our project website.[1] All sequences were captured using a hand-held Canon HV20 digital camcorder in Full HD resolution (1440x1080, anamorphic) at 25fps. Since this camera has a CMOS sensor, rolling shutter artifacts are present, particularly in sequence S2.

The shorter sequences S1, S2 and S7 have 95, 100, and 80 frames respectively, and were processed full resolution; whereas the longer sequences S3, S4, S5, S6 have 180, 270, 225, and 220 frames, respectively, and were processed to 960×720 resolution in order to speed up the alignment process. In the short sequences, an alignment was computed for every pair of frames; in the remaining longer sequences, every frame was aligned to the n-th closest frames in time, with $n = 50$. Sequences S1, S2, S4, and S7 have small view point variations and narrow baselines, and sequences S3, S5, and S6 were captured with a view point span of 10–20 degrees around the object of interest. In addition, sequences S3 and S7 have varying focal lengths caused by zooming. These properties are summarized in Table 6.1.

The run-time of the inpainting algorithm ranged from one hour ($n = 50$ candidates) to four hours ($n > 50$ candidates) running in parallel on a frame server with 64 logical processors. Most of the running time was spent

[1] `http://www.mpi-inf.mpg.de/~granados/projects/vidbginp`

during the alignment stage. All sequences were inpainted using identical parameters.

Each sequence shows two or more people moving in front of a static background. The inpainting task consists into removing one of the persons from each sequence, which is marked using a mask. The mask of the person to be removed, and the mask of the remaining foreground objects were created semi-automatically using the implementation of [Bai09] available in Adobe After Effects CS5 [Adobea] (see Fig. 6.8a).

6.4.2 Discussion of the Inpainting Results

The performance of the inpainting algorithm depends on how well the assumptions of the algorithm are satisfied on each sequence. The inpainting results are presented in Fig. 6.8c. In sequences S1 and S2, the scene has two dominant planes, i.e., the ground, and the facade, and sequence S1 has an additional plane located at the ornament plant. Furthermore, the facade contains non-trivial geometry, such as lamps, doorways, and showcases. Although the geometry in the scene is not completely planar (e.g. the trees and the features in the facade), the proposed algorithm produced a perceptually plausible inpainting. This is possible as the relatively small depth changes of the non-planar objects can be approximated using homographies whenever the disparity between views is also small, as it is the case in these sequences. Additionally, the algorithm enforces consistency across spatial neighborhoods, which minimizes the inclusion of objectionable artifacts in the resulting video.

In sequence S3, the scene has a relatively uniform background, with three dominant planes (the ground and two facades). The frames of this sequence suffer from motion blur due to low illumination. This sequence is challenging for the proposed algorithm due to the lack of distinctive features in the uniform background, which is exacerbated by the motion blur. In this situation, feature point extraction and matching performs poorly, and the resulting homography candidates do not properly represent the geometry of the scene. In addition, the right-hand side of the sequence shows an object with thin structures (a wood fence) that is not contained within any of the three dominant planes. Since no homography was estimated for this structure, the inpainting estimated for this object was not plausible.

The scene in sequences S4–S7 contains two feature-rich dominant planes (the ground and the facade). In this setting, the proposed algorithm produced plausible inpaintings. All the inpainting results are available in our project web page.[2]

[2] `http://www.mpi-inf.mpg.de/~granados/projects/vidbginp`

Issues Caused by Deficient Input Masks

The inpainting results can contain artifacts that are not caused by the inpainting algorithm itself, but rather by inaccuracies in the mask of the object to be removed. For instance, in sequences S4–S6, a faint moving shadow is still visible in the inpainted result whenever the mask does not cover completely the diffuse shadow of the person removed. This artifact is visible although gradient-domain fusion removes illumination discrepancies in the inpainted region. These artifacts are easily detected by humans, since the location of the shadow changes over time, and the human visual system is very sensitive to temporal changes [Wandell95]. This type of artifacts could be corrected by additional refinements of the input mask, but this is a challenging task given that the boundary of diffuse shadows is difficult to localize even with the help of human interaction. This issue could be addressed in the future, for instance, by designing methods for simultaneous alignment and motion segmentation that take into account this type of luminance differences.

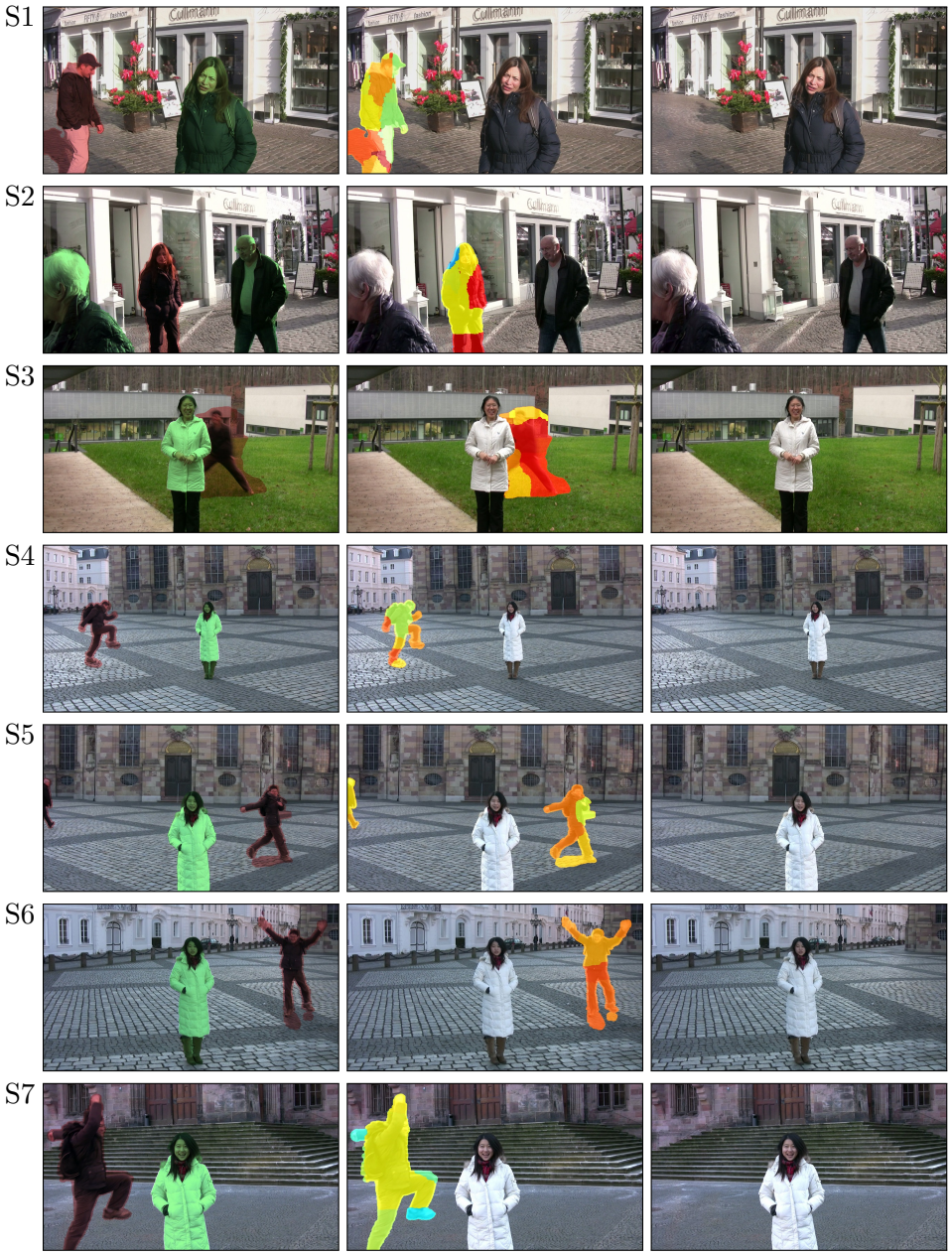
Issues Caused by Semi-transparent Objects

A different type of artifact is caused by the type of representation used for the input masks. Any semi-transparent object (e.g. hair) that is blended with the object to be removed will either be filled with background, or kept as it is (i.e., blended), since the object to be removed is represented as a binary mask. To address this issue, a layer separation algorithm could be applied (e.g. [Yin07]), such that each individual layer could be inpainted separately. However, layer separation is a difficult task, and the success of the inpainting will depend on the quality of the separation.

Issues Caused by Temporal Incoherence

Lastly, despite the temporal consistency term introduced in the Poisson blending (Eq. 6.8), temporal inconsistencies may appear in some sequences (e.g. in sequence S3). This occurs since the proposed method does not directly enforce temporal coherence during the construction of the inpainting composite (Sec. 6.3.2). An alternative to address this issue is to jointly estimate homographies across multiple views [Zelnik-Manor02], in a way that resulting homographies are consistent for several contiguous frames.

Despite the aforementioned shortcomings, the proposed method can generate high quality inpaintings on sequences that do not significantly deviate from the initial assumptions, i.e., piece-wise-planar geometry, and non-flat textures. These assumptions cover a large range of scenes configurations, and in addition, allow the proposed method to be applied in settings where other methods cannot, as demonstrated in the next section.



(a) Input+mask (b) Color-coded sources (c) Inpainting result

Figure 6.8: Inpainting results in the seven test sequences: (a) An input frame and the mask for the hole and dynamic objects (shaded in red and green, respectively); (b) visualization of the source frames obtained by minimizing Eq. 6.8, where each source is shown in a different color; (c) final inpainting result after compositing and gradient-domain fusion.

(a) Input (b) Without alignment (c) Proposed method

Figure 6.9: Example of the artifacts produced by the method proposed in Chapter 5 when applied to videos with camera motion (circled in red). In particular, this sequence shows camera shake and variable focal length. The method proposed in this chapter explicitly compensates for camera motion, and therefore, it can achieve more plausible inpaintings.

6.4.3 Comparison with Alternative Approaches

First, the importance of correcting perspective distortion is illustrated. For this purpose, a comparison was performed with the method presented in Chapter 5, using sequence S7, which has a very narrow camera motion. As shown in Fig. 6.9b, this method cannot produce plausible results, since it does not account for camera motion, and redundant perspective distortions of the same object are unlikely to appear in videos with moving cameras. For this reason, the result produced by the method presented in Chapter 5 was not geometrically consistent.

Second, the limitations of inpainting methods based on structure-from-motion [Bhat07] are illustrated. The method of Snavely et al. [Snavely06] (which is applied in [Bhat07]) was used to reconstruct the camera projection of the sequences S1–S7 presented in this section. However, this method could not produce a correct calibration on any of the sequences. This could be attributed to the narrow baseline, variable focal length, and moving objects that are common to most of these sequences.

Fig. 6.10 illustrates this calibration problem for the sequence S7. This problem visible as the reconstructed geometry is essentially planar even though there are two dominant planes in the scene. In addition, some of the estimated camera positions lie behind the point cloud representing the geometry of the scene. These problems prevented a subsequent application of a multi-view stereo algorithm for reconstructing the depth of each frame, which would allow the rendering of the missing regions from other viewpoints, as proposed in [Bhat07]. As shown in Fig. 6.9c, the proposed method requires less geometrical information about the scene, and therefore, it can provide a plausible inpainting in this case.

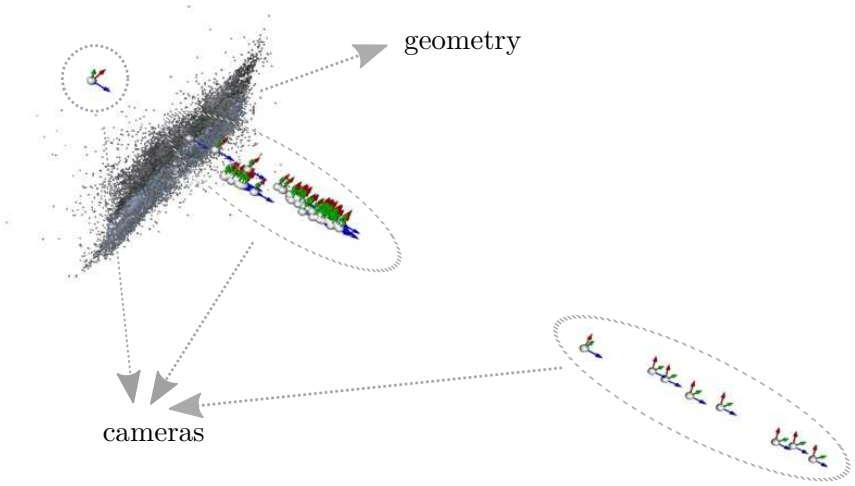


Figure 6.10: Example of incorrect camera projection estimation using a structure from motion method. This estimation was performed on sequence S7 (see Fig. 6.9), which has a narrow baseline and variable focal length. The reconstructed geometry for this sequence, shown as a colored point cloud, is almost planar. Additionally, some of the cameras were estimated to be located behind the scene, even though there was little camera motion in the sequence.

6.5 Conclusion

This chapter proposes a method for removing objects from videos filmed with a moving camera. The objects are removed by replacing the region they occupy with a composite of the other video frames where the background behind them is visible. However, due to perspective distortions induced by camera motion, the appearance of the background might change from frame to frame; such distortions need to be corrected prior to the compositing. This correction is performed independently for every frame pair using a set homographies following the assumption that the scene can be approximated using piece-wise planar geometry. Since the correction is performed between pairs of frames independently and without requiring to recover a global scene and camera models, the proposed method is flexible enough to handle general camera motion and arbitrary scene geometry. This makes it applicable to a wider range of inputs including videos with rotation-only camera motion, narrow-baselines, and variable focal lengths. This is a crucial advantage over previous methods which require applying non-trivial algorithms for recovering the camera projection matrix and scene depth for every frame of the input video.

In the proposed pipeline, the process of correcting for perspective distortions between frame pairs and the process of constructing a composite of the background region are defined as global energy minimization problems. The formulation is equivalent to a Bayesian inference problem using a Markov-random-field prior, and it is solved efficiently using graph cuts for every frame. A final post-processing step is applied for reducing potential illumination mismatches using gradient-domain fusion.

The resulting method is the first in the literature to enable video inpainting of background regions under general camera motion and general scene geometry.

CHAPTER 7

Conclusions

It is increasingly possible to perform editing operations on images and videos that modify not only the appearance but also the semantic content of the recorded scene. This is done in a way that the result looks plausible to humans, but without having to construct models for the actual components of the scene. These components include the geometry, motion, light sources, camera, or materials of the scene. This thesis presented automatic editing methods for modifying the content of the scene without requiring scene models. These methods are developed for two different scenarios: The removal of ghosting artifacts from HDR images reconstructed from image sequences, and the removal of objects from video sequences with static and moving cameras.

This general type of editings are commonly used in industries such as advertisement, publishing, and television and movie production, but are performed by visual artists with different degrees of manual intervention. The degree of manual intervention depends on the availability of automatic or semi-automatic computer vision methods to perform these tasks. However, automatic methods are not available for several kinds of editing tasks, and therefore, many human-hours are spent performing manual editings. The development of such methods often depends on the availability of models for the scene components, which are very difficult to recover from image or videos alone. Therefore, the development of methods that do not require this type of models is highly desirable. The automatic editing methods proposed in this thesis followed this direction.

In order to perform plausible editings without requiring scene models, the ghost-removal and object-removal operations proposed in this work are

defined within an energy minimization framework. In this framework, the editing requirements of the task at hand, together with the plausibility requirements, are expressed in a scalar energy function that determines the amount by which a given editing solution deviates from the requirements. For expressing this deviation in a single scalar, the main task is to find measurable visual properties that are found in the result whenever the requirements are fulfilled. Once these properties are encoded in a energy function, the result is obtained by finding solutions that have low energy.

The main optimization method used in this work is the expansion-move algorithm. This algorithm is a combinatorial optimization method that provides warranties on the maximum distance between the energy of the global minima and of the solutions produced. Other optimization strategies, such as maximum likelihood estimation, and solvers for linear systems of equations were also applied.

Naturally, the visual properties needed for approximating the editing requirements depend on the particular task. The selection of these properties and the definition of the corresponding energy functions, constitute the main contribution of this work. These properties are described separately for the editing of exposure and video sequences in the following sections.

7.1 Editing of Exposure Sequences

7.1.1 Ghosting Removal from Exposure Sequences

In this first task, an image sequence of a dynamic scene is averaged in order to obtain an image with a higher dynamic range (HDR). The editing task consists in detecting regions where moving objects appear in the scene, such that they can be excluded from the average. In this way, the resulting HDR image is free of ghosting artifacts. For this task, a test was designed to verify that the color measurement at the same pixel location in two images corresponds to the same light intensity arriving to the camera sensor. This consistency test is possible if the noise distribution of these color measurements is known. The noise distribution for each image was predicted using a camera noise model, using a new method for calibrating the camera gain factor using arbitrary images. Using the same noise model, this test was extended for verifying sets of images.

This test was encoded in a energy function that encourages the selection of sets of images with as many consistent color values as possible for each pixel, such that the signal-to-noise ratio of the resulting image is maximized. In addition, the plausibility requirements were encoded by testing that the boundaries between adjacent pixels assigned to different subsets of images are also consistent. The resulting deghosting method performed well in highly challenging test scenes. Additionally, the accuracy of the proposed

ghosting detection was found to be the best among existing methods.

7.1.2 Noise-aware HDR Image Processing

The predicted noise distribution for the input images, and for the resulting HDR image was successfully applied to other HDR image processing tasks; these were: Defining an optimal bandwidth for denoising HDR images, determining an exposure sequence for recovering HDR images with a desired minimum signal-to-noise ratio, and defining the optimal weighting function for averaging exposure sequences. In the latter task, the proposed weighting resulted in the best available method for HDR reconstruction.

7.1.3 Future Directions

The noise model used for these editing tasks is applicable to the raw output of digital cameras. A relevant next step is to construct precise noise models that characterize the noise distribution of images in cases when the raw camera output is not available. In this situation, the noise model has to account for the effect of all the transformations occurring during in-camera processing, such as demosaicing, white balancing, dynamic range compression, edge enhancement, and compression. This is a more challenging task since, unlike for imaging sensors, the exact pipeline varies for different camera manufacturers. Liu et al. [Liu08] proposes a method for noise estimation that follows this direction.

If such noise models are available for video cameras, the proposed ghost detection method could be extended to enhance the dynamic range of videos. For instance, in a video taken with a static camera, multiple images of the scene are available. If the static part of these images is averaged, the noise floor of the resulting image is reduced, which effectively extends the dynamic range of the camera. Such an strategy would be ultimately limited by quantization noise, but the dithering caused by shot noise and readout noise helps to overcome this limitation.

Finally, if the camera noise model can be quickly and accurately estimated for arbitrary images without prior calibration, it opens the possibility for extending existing vision algorithms to take advantage of the knowledge of the noise distribution. In particular, the parameters of several vision algorithms could be set according to the predicted noise distribution of the input in order to maximize the expected performance of the algorithm.

7.2 Editing of Video Sequences

7.2.1 Video inpainting on Static Cameras

In this second editing task, an object needs to be removed from a video sequence. The video is captured with a static camera, and it can contain several moving objects. The object is removed by inpainting the appearance of the scene that it occludes, even if the occluded objects are dynamic. The inpainting is performed by computing an offset volume, where each offset points to a location in the video where the missing appearance can be found. This method assumes that the missing appearance can be reconstructed using parts found elsewhere in the video. This is a reasonable assumption given that there is a high degree of redundancy between the frames of a video sequence.

Since every pixel can be inpainted using a different source location, the proposed method tests if the local appearance of the sources of adjacent pixels are compatible, i.e., that their color and gradients are as similar as possible. This test approximates the plausibility requirement put over the inpainting result. This test is performed for every inpainted pixel, and it is encoded in a single energy functional.

The resulting optimization problem is computationally expensive, since there the number of possible sources for each pixel is very large. This set is reduced to cover only relevant source regions with the help of user interaction. The proposed method performed well on several high-resolution sequences of highly complex scenes, and it produced better results than existing competing approaches.

7.2.2 Video Inpainting on Moving Cameras

This task differs from the previous one in that the videos to be edited could contain camera motion. A similar strategy is followed, where the appearance of the scene behind the object to be removed is inpainted using parts of the occluded scene that are visible in other frames of the video. However, before these parts can be used, the perspective distortion caused by camera motion needs to be corrected. This correction is performed following the assumption that the scene can be approximated using piece-wise planar geometry. If this assumption holds, the correction can be performed by computing a set of candidate homographies that align parts of the target and source frames, and selecting the most suitable one for aligning each pixel. The suitability of each homography is encoded in an energy function that tests, for every pixel, whether applying the given homography results in lower color differences with the target frame.

After correcting perspective distortions, there are multiple candidate

sources that can be used to inpaint an occluded pixel, each of them having different degrees of alignment accuracy. As in the previous task, the proposed method tests if the appearance of the sources selected for adjacent pixels are compatible; this test approximates the requirement that the resulting inpainting looks plausible. In addition, the method tests if the alignment of the sources is reliable; this test approximates the requirement that the perspective distortions are well corrected. These tests are performed for every inpainted pixel, and they are combined in a single energy functional. In a post-processing step, remaining illumination mismatches between sources are removed using gradient-domain fusion.

The performance of the proposed method was demonstrated in several real-world video sequences. The resulting inpainting method is the first to work in sequences where the camera undergoes general camera motion, including variations in focal length.

7.2.3 Future Directions

The proposed methods for video inpainting have running times in the order of hours. However, if interactive video inpainting tools were available, they could expand the expression capabilities of artists, and allow them to produce content that is otherwise difficult to create using existing tools. For this purpose, the optimizer used in the proposed methods could be replaced by faster but approximated algorithms. For instance, this could be done by using methods based on random sampling and region growing [Barnes09]. This could compromise the quality of the produced inpaintings, but at the benefit of allowing orders-of-magnitude-faster feedback to the users of the tool.

In a different direction, the method proposed for dynamic inpainting cannot handle objects that suffer scale changes across the video. Similarly, if an occluded motion is available in the video but at a different speed, this source cannot be used as reference for performing the inpainting. This limitation derives from the fact that candidate sources for inpainting are restricted to the current resolution of the video. An alternative for addressing this problem is to consider multiple spatio-temporal scales of the video, such that views of the objects at different spatial and temporal resolutions are available as sources, at the cost of increasing the size of the search space.

An important direction for future investigation is the problem of performing inpainting of dynamic objects in videos with camera motion. Currently, the proposed methods can either inpaint dynamic objects assuming a static camera, or inpaint static objects under camera motion. However, no methods are available for performing inpaintings in the unrestricted case. Naturally, such inpaintings could be produced if a complete model of the scene is available, such that it is possible to render the missing portion of the video; this could be regarded as an editing-by-synthesis framework. A

complete model of the scene should include representations for the geometry of the objects, the material properties, the illumination, and the camera configuration, for the relevant frames of the video. Nevertheless, even in this ideal scenario, the pose of the objects during the occlusion is unknown, and can only be guessed by the inpainting algorithm based on the behavior observed in other instances of the video.

Following this last direction, the proposed methods could be extended to handling stereo video sequences, or monocular sequences where depth estimates are available for each frame. The depth information could be used as an additional condition to test the plausibility of the inpainting results. Alternatively, it could be used to obtain more accurate estimations of the geometry of the scene, so the inpainting algorithms can move towards the direction of methods that utilize stronger scene models.

Bibliography

[Adobea] Adobe. After Effects CS6. 1, 98, 103, 116, 146

[Adobeb] Adobe. Photoshop CS6. 1, 2

[Agarwala04] Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven M. Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael F. Cohen. Interactive digital photomontage. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 23(3):294–302, 2004. 1, 3, 18

[Aguerrebere12] C. Aguerrebere, J. Delon, Y. Gousseau, and P. Musé. Best algorithms for HDR image generation. A study of performance bounds. hal-00733853, version 1, September 2012. 5, 75

[Bai09] Xue Bai, Jue Wang, David Simons, and Guillermo Sapiro. Video SnapCut: Robust video object cutout using localized classifiers. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 28(3), 2009. 1, 3, 98, 103, 116, 146

[Barakat08] Neil Barakat, Thomas E. Darcie, and A. Nicholas Hone. The tradeoff between SNR and exposure-set size in HDR imaging. In *Proc. Intl. Conf. Image Proc. (ICIP)*, pages 1848–1851. IEEE, 2008. 87, 88

[Barnes09] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B. Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 28(3):24:1–24:11, 2009. 1, 37, 61, 97, 121, 124, 140, 157

[Bay08] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc J. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, 2008. 132

[Bell08] André A. Bell, Claude Seiler, Jens N. Kaftan, and Til Aach. Noise in high dynamic range imaging. In *Proc. Intl. Conf. Image Proc. (ICIP)*, pages 561–564. IEEE, 2008. 83

[Bertalmío01] Marcelo Bertalmío, A. L. Bertozzi, and Guillermo Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 355–362. IEEE Computer Society, 2001. 126

[Besag74] J. Besag. Spatial interaction and the statistic analysis of lattice systems. *J. Roy. Statist. Soc.*, 36:192–293, 1974. 10

[Bhat07] Pravin Bhat, C. Lawrence Zitnick, Noah Snavely, Aseem Agarwala, Maneesh Agrawala, Michael F. Cohen, Brian Curless, and Sing Bing Kang. Using photographs to enhance videos of a static scene. In Jan Kautz and Sumanta N. Pattanaik, editors, *Rendering Techniques*, pages 327–338. Eurographics Association, 2007. 126, 128, 135, 144, 149

[Bogoni00] Luca Bogoni. Extending dynamic range of monochrome and color images through fusion. In *Proc. Intl. Conf. Patt. Recogn.*, pages 3007–3016, 2000. 33

[Boykov01] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1222–1239, 2001. 2, 11, 12, 51

[Boykov04] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(9):1124–1137, 2004. 18, 51

[Burt93] Peter J. Burt and Raymond J. Kolczynski. Enhanced image capture through fusion. In *Proc. Intl. Conf. Comp. Vision (ICCV)*, pages 173–182. IEEE, 1993. 32

[Chen02] T. Chen and A. El Gamal. Optimal scheduling of capture times in a multiple-capture imaging system. In M. M. Blouke, J. Canosa, and N. Sampat, editors, *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 4669, pages 288–296, April 2002. 87

[Debevec97] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *Proc. SIGGRAPH*, pages 369–378, 1997. 1, 32, 72, 73, 83

[Debevec98a] Paul E. Debevec. Rendering Synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proc. SIGGRAPH*, pages 189–198, 1998. 31

[Debevec98b] Paul E. Debevec, Yizhou Yu, and George Borshukov. Efficient view-dependent image-based rendering with projective texture-mapping. In George Drettakis and Nelson L. Max, editors, *Rendering Techniques*, pages 105–116. Springer, 1998. 126

[Demanet03] L. Demanet, B. Song, and T. Chan. Image inpainting by correspondence maps: a deterministic approach. *Applied and Computational Mathematics*, 1100:217–50, 2003. 99

[Doe12] Ringo Doe. CHDK: Canon Hack Development Kit, September 2012. 53

[Donoho94] David L. Donoho and Iain M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994. 90

[Drago03] Frédéric Drago, Karol Myszkowski, Thomas Annen, and Norishige Chiba. Adaptive logarithmic mapping for displaying high contrast scenes. *Comp. Graph. Forum (Proc. Eurographics)*, 22(3):419–426, 2003. 53

[Eden06] Ashley Eden, Matthew Uyttendaele, and Richard Szeliski. Seamless image stitching of scenes with large motions and exposure differences. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 2498–2505. IEEE Computer Society, 2006. 3

[Efros99] Alexei A. Efros and Thomas K. Leung. Texture synthesis by non-parametric sampling. In *Proc. Intl. Conf. Comp. Vision (ICCV)*, pages 1033–1038, 1999. 127

[Fischler81] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. 43, 132

[Ford62] L.R. Ford and D.R. Fulkerson. *Flows in networks*. Princeton University, New Jersey, 1962. 18

[Gallo09] O. Gallo, N. Gelfand, W. Chen, M. Tico, and K. Pulli. Artifact-free high dynamic range imaging. In *Proc. Intl. Conf. Computational Photography (ICCP)*, April 2009. 36, 38

[Gallo12] Orazio Gallo, Marius Tico, Roberto Manduchi, Natasha Gelfand, and Kari Pulli. Metering for exposure stacks. *Comp. Graph. Forum (Proc. Eurographics)*, 31(2):479–488, 2012. 87

[Geman88] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. In *Neurocomputing: Foundations of research*, pages 611–634. MIT Press, Cambridge, MA, USA, 1988. 10, 12

[Glasner09] Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *Proc. Intl. Conf. Comp. Vision (ICCV)*, pages 349–356. IEEE, 2009. 98

[Granados08] Miguel Granados, Hans-Peter Seidel, and Hendrik P. A. Lensch. Background estimation from non-time sequence images. In *Proc. Graphics Interface*, pages 33–40, 2008. 37, 38, 110

[Granados10] Miguel Granados, Boris Ajdin, Michael Wand, Christian Theobalt, Hans-Peter Seidel, and Hendrik P. A. Lensch. Optimal HDR reconstruction with linear digital cameras. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 215–222. IEEE, 2010. 5, 32, 71

[Granados12a] Miguel Granados, Kwang In Kim, James Tompkin, Jan Kautz, and Christian Theobalt. Background Inpainting for Videos with Dynamic Objects and a Free-Moving Camera. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *ECCV (1)*, volume 7572 of *Lecture Notes in Computer Science*, pages 682–695. Springer, 2012. 5

[Granados12b] Miguel Granados, James Tompkin, Kwang In Kim, O. Grau, Jan Kautz, and Christian Theobalt. How not to be seen - Object removal from videos of crowded scenes. *Comp. Graph. Forum (Proc. Eurographics)*, 31(2):219–228, 2012. 5, 98, 126

[Granados13] Miguel Granados, Kwang In Kim, James Tompkin, and Christian Theobalt. Automatic Noise Modeling for Ghost-free HDR Reconstruction. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 2013. 5, 39

[Greig89] D. Greig, B. Porteous, and A. Seheult. Exact maximum a posteriori estimation for binary images. *J. Roy. Statist. Soc.*, 51(2):271–279, 1989. 2, 13

[Grosch06] T. Grosch. Fast and robust high dynamic range image generation with camera and object movement. In *Proc. Vision, Modeling and Visualization (VMV)*, pages 277–284, 2006. 35, 38, 65, 66, 67

[Grossberg03] M.D. Grossberg and S.K. Nayar. High dynamic range from multiple images: Which exposures to combine? In *Proc. ICCV Workshop on Color and Photometric Methods in Computer Vision*, 2003. 87

[Hartley04] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, Second edition, 2004. 132, 135

[Hasinoff10] Samuel W. Hasinoff, Frédo Durand, and William T. Freeman. Noise-optimal capture for high dynamic range photography. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 553–560. IEEE, 2010. 32

[HDRSoft] HDRSoft. Photomatrix. 1

[Healey94] G. Healey and R. Kondepudy. Radiometric CCD camera calibration and noise estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(3):267–276, 1994. 19

[Heo10] Yong Seok Heo, Kyoung Mu Lee, Sang Uk Lee, Youngsu Moon, and Joonhyuk Cha. Ghost-free high dynamic range imaging. In Ron Kimmel, Reinhard Klette, and Akihiro Sugimoto, editors, *Proc. Asian Conf. Comp. Vision (ACCV)*, volume 4 of *Lecture Notes in Computer Science*, pages 486–500. Springer, 2010. 36, 65, 66, 67

[Horn81] Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. *Artif. Intell.*, 17(1-3):185–203, 1981. 8

[Hu10] Yiqun Hu and Deepu Rajan. Hybrid shift map for video retargeting. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 577–584. IEEE, 2010. 1, 3, 103

[Jacobs08] Katrien Jacobs, Céline Loscos, and Greg Ward. Automatic high-dynamic range image generation for dynamic scenes. *IEEE Comput. Graph. Appl.*, 28(2):84–93, 2008. 35, 38

[Janesick85] James Janesick. CCD characterization using the photon transfer technique. In K. Prettyjohns and E. Derenlak, editors, *Proc. Solid State Imaging Arrays*, volume 570, pages 7—19. SPIE, 1985. 75

[Janesick01] James Janesick. *Scientific charge-coupled devices*. SPIE Press, 2001. 19, 20, 21, 23, 25

[Jia05] Yun-Tao Jia, Shi-Min Hu, and Ralph R. Martin. Video completion using tracking and fragment merging. *The Visual Computer*, 21(8-10):601–610, 2005. 110

[Jia06] J. Jia, Y.-W. Tai, T.-P. Wu, and C.-K. Tang. Video repairing under variable illumination using cyclic motions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(5):832–839, 2006. 101, 127, 128

[Kang03] Sing Bing Kang, Matthew Uyttendaele, Simon A. J. Winder, and Richard Szeliski. High dynamic range video. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 22(3):319–325, 2003. 33

[Khan06] Erum Arif Khan, Ahmet Oguz Akyüz, and Erik Reinhard. Ghost removal in high dynamic range images. In *Proc. Intl. Conf. Image Proc. (ICIP)*, pages 2005–2008. IEEE, 2006. 37, 38

[Kirk06] Kristian Kirk and Hans Jørgen Andersen. Noise characterization of weighting schemes for combination of multiple exposures. In *Proc. British Mach. Vision Conf. (BMVC)*, volume 3, pages 1129–1138, 2006. 71, 73, 74, 83

[Kolmogorov04] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts? *IEEE*

Trans. Pattern Anal. Mach. Intell., 26(2):147–159, 2004. 13, 15, 16

[Kolmogorov08] Vladimir Kolmogorov, Antonio Criminisi, Andrew Blake, Geoffrey Cross, and Carsten Rother. Probabilistic fusion of stereo with color and contrast for bi-layer segmentation. *Intl. J. Comp. Vision*, 76(2):107, 2008. 3

[Kwatra03] Vivek Kwatra, Arno Schödl, Irfan A. Essa, Greg Turk, and Aaron F. Bobick. Graphcut textures: Image and video synthesis using graph cuts. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 22(3):277–286, 2003. 3, 18

[Larson97] Gregory Ward Larson, Holly E. Rushmeier, and Christine D. Piatko. A visibility matching tone reproduction operator for high dynamic range scenes. *IEEE Trans. Vis. Comput. Graph.*, 3(4):291–306, 1997. 32

[Lensch03] Hendrik P. A. Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 22(2):234–257, 2003. 32

[Ling09] C.-H. Ling, C.-W. Lin, C.-W. Su, H.-Y. M. Liao, and Y.-S. Chen. Video object inpainting using posture mapping. In *Proc. Intl. Conf. Image Proc. (ICIP)*, pages 2785–2788, 2009. 101, 127

[Liu08] Ce Liu, Richard Szeliski, Sing Bing Kang, C. Lawrence Zitnick, and William T. Freeman. Automatic estimation and removal of noise from a single image. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(2):299–314, 2008. 45, 90, 155

[Liu11] Feng Liu, Michael Gleicher, Jue Wang, Hailin Jin, and Aseem Agarwala. Subspace video stabilization. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 30(1):4, 2011. 1

[Lucas81] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In Patrick J. Hayes, editor, *Proc. Intl. Joint Conf. Artif. Intell. (IJCAI)*, pages 674–679. William Kaufmann, 1981. 132

[Mann95] S. Mann and R. W. Picard. Extending dynamic range by combining different exposed pictures. In *Proc. IS&T Ann. Conf.*, pages 442–448, 1995. 32, 72, 73

[Mann01] Steve Mann and Richard Mann. Quantigraphic Imaging: Estimating the camera response and exposures from differently exposed images. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 842–849. IEEE Computer Society, 2001. 32, 72

[Matsushita06] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum. Full-frame video stabilization with motion inpainting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(7):1150–1163, 2006. 102

[Menzel07] Nicolas Menzel and Michael Guthe. Freehand HDR photography with motion compensation. In Hendrik P. A. Lensch, Bodo Rosenhahn, Hans-Peter Seidel, Philipp Slusallek, and Joachim Weickert, editors, *Proc. Vision, Modeling and Visualization (VMV)*, pages 127–134. Aka GmbH, 2007. 36, 38

[Mertens09] Tom Mertens, Jan Kautz, and Frank Van Reeth. Exposure fusion: A simple and practical alternative to high dynamic range photography. *Comp. Graph. Forum (Proc. Eurographics)*, 28(1):161–171, 2009. 32

[Microsoft] Microsoft. Photosynth. 1

[Min09] Tae-Hong Min, Rae-Hong Park, and SoonKeun Chang. Histogram based ghost removal in high dynamic range images. In *Proc. Intl. Conf. Multi. Expo (ICME)*, pages 530–533. IEEE, 2009. 36, 38

[Mitsunaga99] Tomoo Mitsunaga and Shree K. Nayar. Radiometric self calibration. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 1374–1380. IEEE Computer Society, 1999. 32, 72, 73, 85

[Muja09] Marius Muja and David G. Lowe. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In *Proc. Intl. Conf. on Comp. Vision Theory and App. (VISSAPP)*, pages 331–340. INSTICC Press, 2009. 132

[Park11] SungChan Park, Hyun-Hwa Oh, Jaehyun Kwon, Wonhee Choe, and Seong-Deok Lee. Motion

artifact-free HDR imaging under dynamic environments. In Benoît Macq and Peter Schelkens, editors, *Proc. Intl. Conf. Image Proc. (ICIP)*, pages 353–356. IEEE, 2011. 36, 38

[Patwardhan05] Kedar A. Patwardhan, Guillermo Sapiro, and Marcelo Bertalmio. Video inpainting of occluding and occluded objects. In *Proc. Intl. Conf. Image Proc. (ICIP)*, pages 69–72, 2005. 1, 101

[Patwardhan07] K.A. Patwardhan, G. Sapiro, and M. Bertalmio. Video inpainting under constrained camera motion. *IEEE Trans. Image Process.*, 16(2):545–553, February 2007. 102, 127

[Pece10] Fabrizio Pece and Jan Kautz. Bitmap movement detection: HDR for dynamic scenes. In *Proc. Conf. Visual Media Prod. (CVMP)*, pages 1–8, 2010. 36, 38, 65, 66, 67

[Pedone08] Matteo Pedone and Janne Heikkilä. Constrain propagation for ghost removal in high dynamic range images. In Alpesh Ranchordas and Helder Araújo, editors, *Proc. Intl. Conf. Comp. Vision Theory Appl. (VISAPP)*, pages 36–41. INSTICC, 2008. 37, 38

[Pérez03] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 22(3):313–318, 2003. 141

[Pollefeys02a] Marc Pollefeys and Luc J. Van Gool. Visual modelling: From images to images. *J. Vis. Comput. Anim.*, 13(4):199–209, 2002. 1

[Pollefeys02b] Marc Pollefeys, Frank Verbiest, and Luc J. Van Gool. Surviving dominant planes in uncalibrated structure and motion recovery. In Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *Proc. Europ. Conf. Comp. Vision (ECCV)*, volume 2351 of *Lecture Notes in Computer Science*, pages 837–851. Springer, 2002. 126

[Potts52] R. B. Potts. Some generalized order-disorder transformations. *Mathematical Proceedings of the Cambridge Philosophical Society*, 48(01):106–109, 1952. 105

[Pritch09] Y. Pritch, E. Kav-Venaki, and S. Peleg. Shift-map image editing. In *Proc. Intl. Conf. Comp. Vision (ICCV)*,

pages 151–158, Kyoto, Sept 2009. 3, 99, 102, 106, 108, 110, 116, 117, 118, 120, 121

[Pritch11] Yael Pritch, Yair Poleg, and Shmuel Peleg. Snap image composition. In André Gagalowicz and Wilfried Philips, editors, *MIRAGE*, volume 6930 of *Lecture Notes in Computer Science*, pages 181–191. Springer, 2011. 121

[Raman10] Shanmuganathan Raman and Subhasis Chaudhuri. Bottom-up segmentation for ghost-free reconstruction of a dynamic scene from multi-exposure images. In *Proc. Indian Conf. Comp. Vision Graph. Image Process. (ICVGIP)*, pages 56–63, 2010. 36, 38

[Reibel03] Y. Reibel, M. Jung, M. Bouhifd, B. Cunin, and C. Draman. CCD or CMOS camera noise characterization. *Eur. Phys. J.*, 21(21):75–80, 2003. 19

[Reinhard05a] Erik Reinhard and Kate Devlin. Dynamic range reduction inspired by photoreceptor physiology. *IEEE Trans. Vis. Comput. Graph.*, 11(1):13–24, 2005. 53

[Reinhard05b] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec. *High dynamic range imaging: Acquisition, display and image-based lighting.* Morgan Kaufmann Publishers, 2005. 32, 35, 36, 38, 72, 73

[Roberts99] A.J. Roberts. Fast and accurate multigrid solution of Poisson's equation using diagonally oriented grids. *Numerical Analysis*, July 1999. 123

[Robertson03] M.A. Robertson, S. Borman, and R.L. Stevenson. Estimation-theoretic approach to dynamic range improvement using multiple exposures. *J. Elec. Imag.*, 12(2):219–228, 2003. 32, 51, 72, 73, 85

[Rother05] Carsten Rother, Sanjiv Kumar, Vladimir Kolmogorov, and Andrew Blake. Digital tapestry. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 589–596. IEEE Computer Society, 2005. 3, 18

[Schoenemann12] Thomas Schoenemann and Daniel Cremers. A coding-cost framework for super-resolution motion layer decomposition. *IEEE Trans. Image Process.*, 21(3):1097–1110, 2012. 1, 3

[Sen12] Pradeep Sen, Nima Khademi Kalantari, Maziar Yaesoubi, Soheil Darabi, Dan Goldman, , and Eli Shechtman. Robust patch-based HDR reconstruction of dynamic scenes. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(6), 2012. 36, 61

[Shen06] Yuping Shen, Fei Lu, Xiaochun Cao, and H. Foroosh. Video completion for perspective camera under constrained motion. In *Proc. Intl. Conf. Image Proc. (ICIP)*, volume 3, pages 63–66, 2006. 102, 127

[Shi94] Jianbo Shi and Carlo Tomasi. Good features to track. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 593–600, 1994. 132

[Shih09] Timothy K. Shih, Nick C. Tang, and Jenq-Neng Hwang. Exemplar-based video inpainting without ghost shadow artifacts by maintaining temporal continuity. *IEEE Trans. Circuits Syst. Video Technol.*, 19(3):347–360, 2009. 102

[Shiratori06] T. Shiratori, Y. Matsushita, X. Tang, and S. Bing. Kang. Video completion by motion field transfer. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 411–418, 2006. 102

[Shum00] Harry Shum and Sing Bing Kang. Review of image-based rendering techniques. In King N. Ngan, Thomas Sikora, and Ming-Ting Sun, editors, *Proc. Visual Commun. Image Process. (VCIP)*, volume 4067 of *Proceedings of SPIE*, pages 2–13. SPIE, 2000. 126, 144

[Sidibé09] Desire Sidibé, William Puech, and Olivier Strauss. Ghost detection and removal in high dynamic range images. In *Proc. Europ. Signal Process. Conf. (EUSIPCO)*, 2009. 35, 38, 65, 66, 67

[Silk12] Simon Silk and Jochen Lang. Fast high dynamic range image deghosting for arbitrary scene motion. In *Proc. Graphics Interface*, pages 85–92, 2012. 36, 38

[Simakov08] Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. Summarizing visual data using bidirectional similarity. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*. IEEE Computer Society, 2008. 37, 99

[Simoncelli96] Eero Simoncelli and Edward H. Adelson. Noise removal via bayesian wavelet coring. In *Proc. Intl. Conf. Image Proc. (ICIP)*, pages 379–382, 1996. 90

[Snavely06] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: Exploring photo collections in 3D. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 25(3):835–846, 2006. 144, 149

[Srikantha12] Abhilash Srikantha and Desire Sidibé. Ghost detection and removal for high dynamic range images: Recent advances. *Sig. Proc.: Image Comm.*, 27(6):650–662, 2012. 38, 62

[Sun10] Deqing Sun, Stefan Roth, and Michael J. Black. Secrets of optical flow estimation and their principles. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*, pages 2432–2439. IEEE, 2010. 142

[Tomasi98] Carlo Tomasi and Roberto Manduchi. Bilateral Filtering for Gray and Color Images. In *Proc. Intl. Conf. Comp. Vision (ICCV)*, pages 839–846, 1998. 47, 90

[Tomaszewska10] Anna M. Tomaszewska and Mateusz Markowski. Dynamic scenes HDRI acquisition. In Aurélio C. Campilho and Mohamed S. Kamel, editors, *Proc. Intl. Conf. Image Anal. Recogn. (ICIAR)*, volume 2 of *Lecture Notes in Computer Science*, pages 345–354. Springer, 2010. 37, 38

[Torr98] P. H. S. Torr, A. Zisserman, and S. Maybank. Robust detection of degenerate configurations for the fundamental matrix. *Computer Vision and Image Understanding*, 71(3):312–333, 1998. 135

[Torr99] Philip H. S. Torr, Andrew W. Fitzgibbon, and Andrew Zisserman. The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. *Intl. J. Comp. Vision*, 32(1):27–44, 1999. 126

[Tsin01] Yanghai Tsin, Visvanathan Ramesh, and Takeo Kanade. Statistical calibration of the CCD imaging process. In *Proc. Intl. Conf. Comp. Vision (ICCV)*, pages 480–487, 2001. 32, 71, 72, 73, 74, 83

[Veksler99] O. Veksler. Efficient graph-based energy minimization methods in computer vision, 1999. 11, 108

[Veksler10] Olga Veksler, Yuri Boykov, and Paria Mehrani. Superpixels and Supervoxels in an Energy Optimization Framework. In *Proc. Europ. Conf. Comp. Vision (ECCV)*, pages 211–224, 2010. 43

[Venkatesh09] M. V. Venkatesh, S. S. Cheung, and J. Zhao. Efficient object-based video inpainting. *Pattern Recogn. Letters*, 30(2):168–179, 2009. 101, 127

[Wandell95] B. A. Wandell. *Foundations of Vision.* Sinauer Associates, Inc., 1995. 97, 147

[Ward03] Greg Ward. Fast, robust image registration for compositing high dynamic range photographs from handheld exposures. *J. Graphics Tools*, 8:17–30, 2003. 36

[Wexler07] Yonatan Wexler, Eli Shechtman, and Michal Irani. Space-time completion of video. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3):463–476, 2007. 1, 102, 107, 110, 111, 116, 117, 118, 127

[Yin07] Pei Yin, Antonio Criminisi, John M. Winn, and Irfan A. Essa. Tree-based classifiers for bilayer video segmentation. In *Proc. Comp. Vision Patt. Recogn. (CVPR)*. IEEE Computer Society, 2007. 147

[Zelnik-Manor02] Lihi Zelnik-Manor and Michal Irani. Multiview constraints on homographies. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(2):214–223, 2002. 147

[Zhang12a] Wei Zhang and Wai-Kuen Cham. Gradient-directed multiexposure composition. *IEEE Transactions on Image Processing*, 21(4):2318–2323, 2012. 37, 38

[Zhang12b] Wei Zhang and Wai-Kuen Cham. Reference-guided exposure fusion in dynamic scenes. *J. Visual Commun. Image Represent.*, 23(3):467–475, 2012. 36, 38

[Zimmer11] Henning Zimmer, Andrés Bruhn, and Joachim Weickert. Freehand HDR imaging of moving scenes with simultaneous resolution enhancement. *Comp. Graph. Forum (Proc. Eurographics)*, 30(2):405–414, 2011. 33, 61