A HYBRID CAMERA SYSTEM FOR LOW-LIGHT IMAGING

by

Feng Li

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science

Fall 2011

A HYBRID CAMERA SYSTEM FOR LOW-LIGHT IMAGING

by

Feng Li

Approved: _____
Errol L. Lloyd, Ph.D.
Chair of the Department of Computer and Information Sciences

Approved: _____
Babatunde A. Ogunnaike, Ph.D.
Interim Dean of the College of Engineering

Approved: _____
Charles G. Riordan, Ph.D.
Vice Provost for Graduate and Professional Education

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Jingyi Yu, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Chandra Kambhamettu, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Christopher Rasmussen, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Rob Fergus, Ph.D.
Member of dissertation committee

ACKNOWLEDGEMENTS

It is with immense gratitude that I acknowledge the support and help of my research advisor Jingyi Yu. It was him who brought me into this exciting research area of computational photography and computer vision. He not only provides me with the freedom to find my own way in the research but also guides me to stay on track. He is always available to me and willing to spend enormous of time to mentor me. He is the type of advisor that every graduate student wants.

My sincerest gratitude goes to Chandra Kambhamettu, Christopher Rasmussen, and Rob Fergus for serving on my advisory committee. They have given me invaluable comments and suggestions on both my study and my research, lent moral support, and provided wise career advisement.

I share the credit of my work with collaborators at the University of Delaware and other research institutes. They are Jinxiang Chai, Jian Sun, Jue Wang, Philippe Guyenne, David Saunders, Haibin Ling, Xuan Yu, Yuanyuan Ding, Liwei Xu, Zhan Yu, Yu Ji, Christopher Thorpe, Scott Grauer-Gray, Yi Wu, and Zijia Li. It has been a great privilege to work with each of them, and this dissertation would have remained a dream had it not been for their constant help and invaluable discussions.

I would like to thank Microsoft Research Asia, Technicolor and MERL for providing me the wonderful internship experience during my Ph.D. study. I consider it an honor to work with my co-authors and mentors in these labs: Jian Sun, Izzat Izzat, and Fatih Porikli. I want to thank them for taking the time to share their expertise and knowledge of the field. It was through these times that I learned tremendous amount from every one of them.

I would also like to thank other members of the UD graphics lab (Liang Wei, Kevin Kreiser, Luis D. Lopez, Miao Tang, Yuqi Wang, Jinwei Ye, Xinqing Guo and Xiaogang Chen) for sharing their thoughts, codes and providing such an enjoyable and supportive

environment during my Ph.D. study. Thanks to Xiaozhou Zhou for the valuable suggestions about my submission of fluid-type motion estimation. Special thanks to Li Jin for her timely help on my dissertation and presentations. Thanks to all of my friends in UD, I enjoyed the time that I spent with everyone of you.

I owe my deepest gratitude to my wife and my parents. Thanks to them for being proud of me and loving me, and all the sweetest memories.

DEDICATION

I dedicate this dissertation to my loving wife whose encouragement have meant to me so much during the pursuit of my Ph.D. degree. I dedicate this dissertation to my parents who have given me support throughout my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LISF OF ALGORITHMS

ABSTRACT

Acquiring high quality imagery in darkness has been a challenging problem in computer vision. In this dissertation, we propose to develop a new imaging system that is suitable for low light conditions. Our proposed solution is to integrate multispectral, multi-speed, multi-resolution, and multi-focus sensors into a hybrid camera array. We also develop companion computational photography algorithms for effectively fusing the captured imagery data.

On the system side, I extend the University of Delaware (UD) light field camera array by integrating a broader range of cameras. Specifically, our proposed system will consist of a pair of high-resolution monochrome (HR-M) cameras, a pair of high-speed monochrome (HS-M) cameras, and a single high-resolution RGB color (HR-C) camera. The HR-M cameras further use a wide aperture for acquiring images with low level of noise images whereas the HS-M cameras use fast shutters for capturing motion-blur free images on fast moving objects. Finally, the HR-C camera is used to capture color information about the scene and it uses slow shutters to reduce the color noise.

On the algorithm side, I develop a class of novel algorithms that combine the capabilities of classical computer vision and computational photography for fusing the imagery data from the hybrid sensors. First, I propose to develop techniques for fusing the images from the HR-M cameras. Since each HR-M camera uses a wide aperture, its images exhibit shallow depth-of-field effects, i.e., it can only focus at one depth layer of the scene. Therefore, we focus the two HR-M cameras at different scene depth and design multi-focus multi-view fusing methods for synthesizing all-focus video streams.

Next, I use the HR-M imagery data as the multispectral prior to denoise the HS-M streams. We first preprocess the HS-M images to improve their contrast using a virtual exposure fusion technique. We then locate the corresponding patches on the HR-M images.

Finally, to denoise the HS-M image, I design an alternating optimization algorithm by extending the Total Variation (TV) denoising algorithm by imposing the gradient fields of the corresponding HR-M patches as priors.

Recall that the HR-C camera uses a slow shutter and therefore will incur severe motion blurs on moving targets. To resolve this problem, I use the denoised HS-M frames for estimating the blur kernel on the HR-C camera and then apply deblurring techniques to reduce/remove motion blurs. We have applied our hybrid camera array system to capture both indoor and outdoor scenes under low light. Experimental results show that our solution can greatly enhance the quality of the imagery by reducing noise, improving image sharpness, and removing motion blurs.

Chapter 1

INTRODUCTION

In recent years, the use of high resolution, high speed, high dynamic range, or multi-spectral cameras have become a common practice in many imaging applications. However, thus far no single image sensor can satisfy the diverse requirements of all industrial camera applications today. In particular, it remains an open problem to design suitable sensors for conducting surveillance tasks such as tracking, detecting, and identifying targets in highly cluttered scenes in low-light conditions. When using digital cameras for such tasks, it is important to increase the exposure time or sensor sensitivity to capture sufficient light. Increasing the exposure time, however, would introduce motion blurs for moving objects and lead to significant degradation to image quality while increasing the sensor sensitivity would exaggerate the ambient random noise fluctuations, making image denoising more challenging.

In theory, each individual problem in low light imaging could be tackled by changing the settings of aperture, shutter speed, focus, sensor wavelength, and etc. For example, high-speed cameras can capture fast motion with little motion blur, but require expensive sensing, bandwidth and storage. The image resolution in these cameras is often much lower than many commercial still cameras. This is mainly because the image resolution has the linear inverse relationship with the exposure time [7] to maintain the Signal-to-Noise Ratio (SNR), i.e., higher speed maps to a lower resolution. In addition, the relatively low bandwidth on usual interfaces like USB 2.0 or FireWire IEEE 1394a also restricts the image resolution especially when streaming videos at 100 to 200 frame/second.

To guarantee enough exposures, one could choose to use either a wide aperture or a slow shutter. For example, if we couple a wide aperture with fast shutters, we shall be able to

Table 1.1: Comparisons between different camera sensors

Type	Advantages	Disadvantages
High Speed	fast shutter	low image resolution require large data bandwidth
High Resolution	rich spatial details	blurs for fast motions
Multispectral	high contrast high dynamic range	require special equipment

capture low-noise imagery and fast motions. However, wide apertures lead to shallow depth-of-field (DOF) while only parts of the scene can be clearly focused. In contrast, by coupling a slow shutter with a narrow aperture, one can capture all depth layers in focus. However, slow shutters are vulnerable to fast moving objects which will cause severe motion blurs in the acquired images.

Overall, high speed sensor is capable to capture fast moving objects without motion blurs. High resolution sensor can provide fine detail of the scene. Multi-spectral sensors can capture wavelengths beyond the visible light that are potentially advantageous to reducing noise and enhancing contrast. However, no sensor by far is perfect: each type of the sensors has its disadvantages, as shown in Tab. 1.1.

Recent advances in digital imaging suggest that it may be possible to construct hybrid imaging devices by combining a heterogeneous type of sensors. In this dissertation, we present a new imaging system that integrates multi-spectral, multi-speed, multi-focus, and multi-resolution sensors into a hybrid camera array and use this new imaging device for conducting surveillance tasks under low light conditions. We further develop the companion computational photography algorithms for effectively fusing the imagery data from the heterogenous types of sensors.

1.1 Dissertation Statement

Our research explores new imaging systems to improve the effectiveness and robustness in surveillance under poor lighting conditions. By leveraging multi-spectrum, high-speed, and high-resolution image sensing, we develop a hybrid camera array to satisfy the diverse requirements of surveillance.

Hardware Design:

To tackle the problem of capturing high quality images under low light, we first propose to design a hybrid camera system that combines the advantages of high speed, high resolution, and multi-spectral sensors. Our proposed system extend the University of Delaware (UD) light field camera array by integrating a broader range of cameras. Specifically, our system consist of a pair of high-resolution monochrome (HR-M) cameras, a pair of high-speed monochrome (HS-M) cameras, and a single high-resolution RGB color (HR-C) camera. All monochrome cameras can be equipped with Near Infra-Red (NIR) filters for acquiring high quality imagery data when active NIR light sources are available. In addition, the HR-M cameras further use a wide aperture for acquiring images with low level of noise whereas the HS-M cameras use fast shutters for capturing motion-blur free images on fast moving objects. Finally, the HR-C camera is used to capture color information about the scene and it uses slow shutters to reduce the color noise.

Algorithm Design:

On the algorithm side, we develop new multi-focus fusion, multi-spectral denoising, and multi-speed motion deblurring modules for synthesizing high quality, high speed, and high resolution imagery from the captured data. Specifically, we explore the rich information captured by the hybrid camera, in both spatial and temporal domains, for designing robust algorithms for low light imaging.

Fig.1.1 shows the processing pipeline for the low-light imaging system using our hybrid camera. We first develop new techniques for fusing the images from the HR-M cameras. Since each HR-M camera uses a wide aperture, its images exhibit shallow depth-of-field effects, i.e., it can only focus at one depth layer of the scene. Therefore, we propose to focus the two HR-M cameras at different scene depth and design multi-focus multi-view fusing

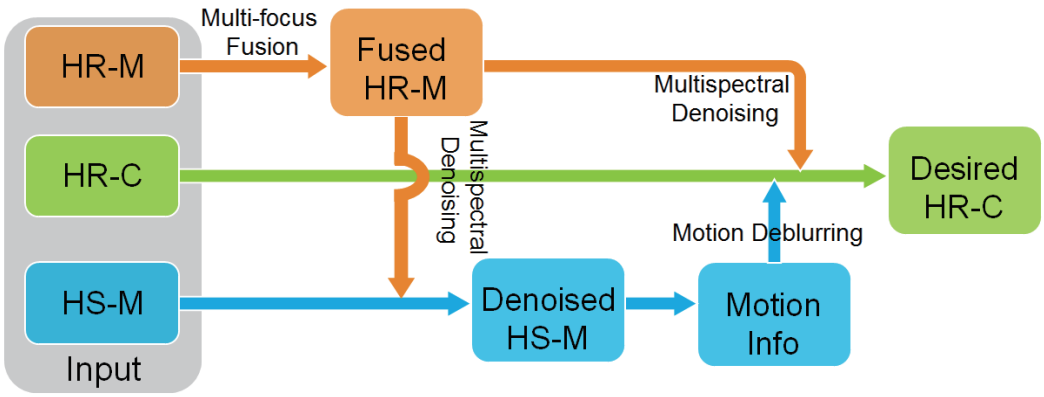


Figure 1.1: System pipeline for low-light imaging.

methods for synthesizing all-focus video streams.

Next, we present an alternating optimization algorithm to remove sensor noise captured by HS-M cameras. We model image denoising as an optimization problem and regularize it by a ℓ_1 total variation (TV) term and a multispectral prior. The TV regularization can remove unwanted details while preserving important details such as edges. Our multispectral prior can bring missing details from the fused all-focus HR-M image to the low resolution noisy HS-M images. These terms both regularize the objective function in this minimization problem to effectively reconstruct low-noise HS-M image sequences.

Recall that the HR-C camera uses a slow shutter and hence will incur severe motion blurs on moving targets. We use the denoised images from the HS-M cameras for estimating blur kernels on the HR-C camera. We then apply deblurring techniques such as Richardson-Lucy deconvolution to reduce the motion blur.

1.2 Contributions

This dissertation makes the following contributions in system designs and algorithm developments.

System Design:

- A UD light field camera array system that uses one workstation to support a 3×3 camera array capturing synchronized, well-calibrated, and uncompressed video sequences at 1024x768x8bit@30fps or 1024x768x24bit@15fps. The maximum data recording bandwidth of this system is up to 500MB/s.
- We further construct a hybrid camera array that combines the advantages of high-speed, high resolution, and multispectral image sensors. All sensors in our camera array are synchronized through software solutions and they are controlled by a single workstation.

Algorithm Developments:

- We develop a novel multi-focus fusion technique for low light imaging. It captures an dual focus stereo pair using wide apertures to reduce sensor noises, and then fuses an all-focus image for multi-spectral denoising of HS-M and HR-C cameras.
- We develop an alternating optimization algorithm to remove sensor noise in HS-M images. We first preprocess the captured images using a new virtual exposure technique, then conduct patch matching between the HR-M image pairs and the HS-M images. We use these patch priors and a ℓ_1 total variation term to regularize the objective function for image denoising.
- Finally, we develop a robust algorithm for motion deblurring and depth map super-resolution. Our method first recovers a low resolution depth map from the HS-M pair and combines it with each HS-M's motion flow to estimate the point spread functions (PSFs) in the HR-C image. After motion deblurring the HR-C image, we then use it to upsample the low resolution depth map using joint bilateral filters.
- We apply our hybrid camera and algorithms for capturing both indoor and outdoor scenes under low light. Our new imaging system is capable of reconstructing high quality color images at 1024x768x24bit@7.5fps and can be used as basis for other low light applications, such as pedestrian detection, recognition, and object tracking.

1.3 Blueprint of the Dissertation

This dissertation is organized as follows. Chapter 2 describes the related work on camera array designs and defocusing, deblurring and denoising. Chapter 3 describes the hardware design of our hybrid camera array. Chapter 4 – 6 describes our computational photography algorithms for generating high resolution, low-noise, low motion blur color video streams under low-light conditions. Specifically, Chapter 4 describes our multi-focus fusion

technique. Chapter 5 describes the multispectral denoising framework. Chapter 6 discusses motion deblurring. Chapter 7 demonstrates combining the hardware and the algorithms for conducting low-light surveillance. Chapter 8 concludes the thesis and discusses future extensions.

Chapter 2

RELATED WORK

Recent advances in digital imaging and computer vision [95] has led to the developments of many new imaging systems and computational photography algorithms. The use of camera array [49, 54, 74, 147], specially designed apertures [5, 42, 69, 121], lens [83], flash [64, 94], or shutters [93] has become a common practice for various imaging applications. In this chapter, we briefly review existing computational imaging approaches for multi-modal fusion that are closely related to this work.

2.1 Multi-Camera System

In recent years, a number of camera systems have been developed for specific imaging tasks. For example, the Stanford light field camera array [73, 120, 127, 128] is a two dimensional grid composed of 128 1.3 megapixel firewire cameras which stream live videos to a stripped disk array. Wilburn et al. [128] showed that this large camera array can be treated as a single camera and applied it for high speed imaging, high-resolution imaging, and wide aperture photography with dynamic depth-of-field control. The MIT light field camera array [134] uses a smaller grid of 64 1.3 megapixel USB webcams for synthesizing dynamic Depth-of-Field effects. Zitnick et al. [147] developed a multi-view video capture system with 8 synchronized video cameras mounted along a horizontal rig. Their goal is to use computer vision methods to reconstruct the 3D scene and then synthesize new views from the recovered geometry and the input videos. McGuire et al. [81] combined 3 cameras with varying focus and aperture settings to automatically extract a matte image. They elaborately design the system using beamsplitters so that all three cameras can capture the scene from the same viewpoint. Along the same line, Joshi et al. [54] constructed a linear array of 8 Basler cameras for natural video matting. Their setting is more flexible as the cameras no

longer need to have the same viewpoint. They first create a synthetic wide aperture image with focused foreground and defocused background and then extract the alpha matte.

Instead of using multiple cameras, it is also possible to manipulate the light rays using lenslet arrays or attenuating masks [38, 83, 121]. The key idea here is to trade spatial resolution for angular resolution: cameras these days have very high resolution and if we can distribute the resolution into multiple views, we can effectively capture a light field. For example, Ng et al. [83] designed a light field camera that combines a single DSLR with a microlenslet array. Each lenslet captures the scene from a different viewpoint and the lenslet array effectively emulates a camera array. By using a large number of densely packed lenslets, one can virtually capture a large number of viewpoints in a single image. It also, however, comes at the cost of reduced resolution for each light field view. A light field camera is typically paired with an ultrahigh-resolution static DSLR and is therefore not applicable to video streaming. Liang et al. [77] proposed a programmable aperture scheme to capture light field at full sensor resolution through multiple exposures without any additional optics and without moving the camera. However, they require both the scene and the camera be static as well since their data are captured sequentially. To improve the spatial resolution of light fields captured by an plenoptic camera, Georgiev and Lumsdaine [39] proposed the focused plenoptic camera model for light field super-resolution.

Similar to the lenslet based multi-view acquisition system, mirror-based (catadioptric) systems can also be used to acquire the incident light fields. In essence, these systems point a commodity perspective camera at an array of spherical or hyperbolic mirrors to simultaneously capture multiple views. Their applications include image-based relighting [119], triangulation of a point light source [67], object detection for video surveillance [58], and 3D reconstruction [24, 65]. The major drawback of catadioptric systems is that the images captured are often multi-perspective due to reflection distortions and therefore conventional perspective-camera based vision and graphics algorithms are not directly applicable for processing these images. For example, to reconstruct the 3D scene, Lanman et al. [65] locate the corresponding points in each sphere image via semi-automatic approaches. Ding et al. [24] proposed to use general linear cameras (GLC) [139] to approximate each mirror image as

piecewise primitive multi-perspective cameras for stereo matching and volumetric reconstruction. Taguchi et al. [111] recently developed a closed-form solution for computing the projection from a 3D point to its image in axial-cone based mirrors for efficient view synthesis.

The major difference between these existing multi-camera/light field systems and our proposed solution is that we use a heterogenous set of cameras, i.e., cameras in the system have different aperture, shutter, resolution, or spectrum settings.

2.2 Low Light Imaging: Denoising vs. Defocusing

The main application of our system is high quality low-light imaging. Acquiring high quality imagery under low light using a commodity camera has been a challenging problem in computer vision and computational photography. There are essentially two approaches to reduce noise at the capture stage: extend the exposure time or increase the aperture size.

2.2.1 Aperture

Wide apertures allow more light to be admitted to the camera and are suitable for low-light and fast motion imaging. However, they also lead to shallow depth-of-field (DOF) where pixels are sharp around what the lens is focusing on and blurred elsewhere. Most existing methods have been focused on effectively using defocusing for recovering scene geometry as well as designing tools for reducing defocus blurs.

Depth-from-Defocusing (DfD). Recovering scene depth from multiple defocused images is a well-explored problem in computer vision. Most existing approaches require capturing a large number of images from the same viewpoint with different focuses [35, 56]. By minimizing the blur, scene depth can then be estimated. For example, [45] captures all possible combinations of aperture and focus settings to recover very high quality depth maps. It is also possible to fuse the estimated DfD depth map with stereo map, e.g., via weighted fusion [3]. [92] uses two stereo pairs, each having a different focus, to separately model the depth map and the defocused image as Markov random fields. Finally it is also possible to

conduct a temporal defocus analysis method to recover depth map by estimating the defocus kernel of the projector [143].

Specially Designed Apertures. In the emerging field of camera imaging, several researchers have suggested that replacing regular circular-shaped apertures with specially designed ones has many benefits. For example, coded apertures [69, 121, 146] change the frequency characteristics of defocus blur and use special deconvolution algorithms to reduce blurs and estimate scene depth from a single shot. Color-filtered aperture [5] divides the aperture into three regions through which only light in one of the RGB color bands can pass. Out-of-focus regions will lead to depth-dependent color misalignments, which can then be used to determine scene depth. For more details, we refer the readers to the survey of computational photography [95]. A downside of these approaches is the requirement of modifying the camera optical system.

Multi-focus Fusion. It is also possible to create all-focus images using multi-focus photomontage. Levin et al. [69] uses coded apertures coupled with special deconvolution algorithms based on sparse-prior to recover an all-in-focus image for refocusing. Since they use regular circular-shaped apertures, their technique cannot fully recover the high-frequency components in the defocused images. Alternatively, digital photomontage systems [1] provide users with an interactive interface to highlight the desired regions in different images and then automatically fuse the selected regions. Dai and Wu use image matting techniques to iteratively recover the foreground and background layer from a partial blur image [22]. It is also possible to capture a sequence of images from the same viewpoint with varying focus and then merge them to synthesize extended depth of field [44]. These systems either rely on user inputs and graph-cut optimization to segment the desired regions, or requires viewpoint registrations.

2.2.2 Shutter

An alternative approach to increase the exposure is to use a slow shutter. However, slow shutters will introduce motion blurs when capturing fast moving objects. Majority

efforts have hence been focused on robust image deblurring algorithms.

Image Deblurring. The problem of image-deblurring has been well studied in the image processing community, since taking satisfactory photos of fast moving objects under low light conditions using a hand-held camera is very challenging. Most previous methods have focused on recovering the blur kernel, or the point spread function (PSF). The classical Wiener filter [138] is usually used to restore the blurred image when PSF is given or estimated. Computer vision methods such as graph cuts [91] and belief propagation [115] have also been used to recover nearly optimal deblurred images. Tull and Katsaggelos [118] proposed an iterative restoration approach to further improve the quality of deblurred images. Jia et al. [50] improved a short exposure noisy image by using color constraints observed in a long exposure photo without solving for the PSF. Fergus et al. [34] used a zero-mean Mixture of Gaussian to fit the heavy-tailed natural image prior, and employed a variational Bayesian framework for blind deconvolution. Yuan et al. [141] show that a sharp and noisy short exposure of a scene and a long exposure of the same scene can be combined together to reconstruct a PSF and a high-quality latent image.The advantage of this method is that the denoised image can be used as a good initial guess for the latent image L estimation. Furthermore, during the deconvolution step of their algorithm, they use the denoised image as prior which increases the quality of the deconvolution. Most single-image deblurring algorithms assume the PSF is spatially-invariant. An exception is the work by Levin [68] in which the image is segmented into several layers with different kernels, each assumed to have a constant motion velocity.

More recent image deblurring techniques have focused on using priors for guiding the deconvolution process [51, 63, 72, 102, 104, 133]. Shan et al. [102] exploited sparse priors for both the latent image and blur kernel and developed an alternating-minimization scheme for blind image deconvolution. Levin et al. [72] showed that common MAP methods based on estimating both the image and kernel are not reliable and often lead to trivial solutions. In contrast, estimating the blur kernel is better constrained since the kernel has much smaller size than the image. Xu and Jia [133] proposed an efficient kernel estimation method based

on spatial priors and strong edges exhibited in the image. Other types of priors include total variation regularization (also referred to as Laplacian prior) [19, 114, 123], heavy-tailed natural image priors [34, 102], color priors [53], and Hyper-Laplacian priors [63].

Hardware Solution. Hardware solutions have also been proposed to reduce motion blurs. These include the techniques based on lens stabilization and sensor stabilization. For example, adaptive optical elements controlled by inertial sensors have been used to compensate for camera motion [18, 84]. Joshi et al. [52] used a combination of inexpensive gyroscopes and accelerometers in an energy optimization framework to estimate a blur function from the camera's acceleration and angular velocity during an exposure. Coupled with natural image prior, they proposed to deblur the images via a joint optimization scheme.

Our work is inspired by the hybrid speed camera by Ben-Ezra and Nayar [7]. To estimate blur kernels for image deconvolution, they developed an imaging system that consists of a low resolution high speed (LRHS) and a high resolution low speed (HRLS) camera.They assume that motion blurs are caused by the shaking of the camera and they track the motion in the LRHS camera to recover the PSF and then deblur the image. It is also possible to apply their system to deblur moving objects [8]. However, their method only supports spatially invariant blur kernels. As an extension to [8], Tai et al. [112] recently constructed a hybrid camera system to reduce spatially-varying motion blur in videos and images. They attach a LRHS camera to a HRLS camera, align their optical axes using a beamsplitter, and apply an iterative algorithm to remove blurs.

The major difference between these existing computational imaging solution and ours is that we do not assume that the cameras are co-axial. Although this co-axial setup could give accurate registrations between different viewpoints, it requires special equipments to precisely calibrate the system, and splits the incident rays accross different views, which makes it not suitable for low light applications. We mount the heterogenous sensors on a 2D array and actively use the scene depth information (e.g., disparity and focus variations) across the cameras.

Image Denoising. Finally, it is also possible to directly process the noisy image via

denoising algorithms. There is a considerable amount of literature on image denoising. Here we only focus on recent computational imaging solutions and the state-of-the-art patch-based techniques. On the computational photography front, the flash/non-flash pair photography [87] aims to use the image captured under flash to enhance the one captured without flash, e.g., via bilateral filters [85, 117]. Bennett et al. [9] demonstrated a per-pixel exposure model to enhance underexposed visible-spectrum video. They use the "dual bilateral" to fuse the visible-spectrum videos for temporal noise reduction and color/edge preserving. Krishnan and Fergus [64] captured a pair of images using "dark" flash, that is, low power infra-red and ultra-violet light outside the visible range. They then exploit the correlations between images captured with the dark flash and with ambient illumination. An important aspect of their solution is their optimization technique: they use the strong correlations between color channels to regularize an optimization scheme and then solve it using Iterative Re-weighted Least Squares [69, 106]. It is also possible to use images captured under the same setting but from multiple views for denoising. Zhang et al. [144] captured a sequence of images under low light and then estimate the rough correspondences between the images. To denoise a patch, they first find a stack of similar patches and then use the principal component analysis and tensor analysis to remove image noise. In their implementation, a large number of images ($\sim$ 20) need to be used for robust denoising. Our work, in contrast, aims to use images captured with different settings (aperture, shutter, resolution, and spectrum) and we only use 5 images/cameras.

For single image denoising, it is commonly acknowledged that the patch-based 3D filtering (BM3D) technique [21] is by far the state-of-the-art. Built on the concept of non-local means [14], BM3D exploits self-similarity within an image: it first groups similar patches into a 3D stack and applies hard-thresholding to the transformed coefficients in the 3D wavelet domain, then dispatches each patch of the 3D stack back to its original place to generate the initial estimate, and finally runs this process again but applying the optimal Wiener filtering to the transformed coefficients to reconstruct the denoised image.

Along the same *non-local* direction, i.e., using non-local averaging of all pixels in an image, Chatterjee and Milanfar [20] proposed a locally learned dictionaries framework

to reduce image noise by clustering the given noisy image into regions of similar geometric structure. Recently, Mairal et al. [80] combined the sparse coding with the non-local means method, and proposed to jointly decompose groups of similar signals on subsets of the learned dictionary. Their methods can be used for a range of applications including image denoising, demosaicking, and image completion. Different from these *non-local* denoising methods, *local* methods [16,30,79] aim to construct or learn a dictionary as the basis functions to enforce the sparsity priors commonly observed in the natural images. It is also possible to combine non-local and local methods. For example, Dong et al. [26] formulated the denoising problem as a double-header ℓ_1 optimization problem that is regularized by both dictionary learning and structural filtering. Most recently, Levin and Nadler [71] suggested that state-of-the-art single-image based denoising algorithms are approaching optimality. In this thesis, we borrow the idea of dark flash photography [64] to achieve multi-camera denoising (Chapter 5).

Chapter 3

HYBRID CAMERA SYSTEM DESIGN

In this chapter, we elaborate on the system design of our hybrid camera arrays. We first present a portable multi-view acquisition system by constructing a 3x3 camera array. We analyze some important issues that affect the design of our imaging system, including bandwidth concern, data recording device, camera synchronization, computer architecture, and etc. We also demonstrate using our camera array for dynamic fluid surface acquisitions. Next, we modify the camera array by using heterogenous types of sensors. We carefully choose the appropriate sensors for the task of low-light imaging.

3.1 Light Field Camera Array

In recent year, a number of camera systems have been developed for specific imaging tasks. For example, the Stanford light field camera array [73, 120, 127, 128] is a two dimensional grid composed of 128 1.3 megapixel firewire cameras which stream live video to a stripped disk array. The MIT light field camera array [134] uses a smaller grid of 64 1.3 megapixel USB webcams for synthesizing dynamic Depth-of-Field effects. These systems require using multiple workstations and their system infrastructure such as the camera grid, interconnects, and workstations are bulky, making them less suitable for on-site tasks.

We have constructed a small-scale camera array controlled by a single workstation, as shown in the left of Fig.3.1. Our system uses an array of 9 Pointgrey Flea2 cameras to capture the dynamic fluid surface. We mount the camera array on a metal grid and support the grid through two conventional tripods, so that we can easily adjust the height and the orientation of the camera array. The camera array is connected to a single data server via 4 PCI-E Firewire adaptors. The use of Firewire bus allows us to synchronize cameras through the Pointgrey software solution.

Target Application. The target application of our system is to recover dynamic 3D fluid surfaces. To do so, we place a known pattern beneath the surface and position the camera array on top to observe the pattern. By tracking the distorted feature points over time and across cameras, we obtain spatial-temporal correspondence maps and we use them for specular carving to reconstruct the time-varying surface. In case one of the cameras loses track due to distortions or blurs, we use the rest cameras to construct the surface and apply multi-perspective warping to locate the lost-track feature points so that we can continue using the camera in later frames. We apply our system to capture a variety types of fluid motions.

This system setup gives us many advantages for multi-view data streaming:

- Low maintainance. Because of the single workstation and compact camera grid design, we don't need to develop distributed control modules, sophisticated data saving software, and complicated multi-camera calibration algorithms. Thus our system is very easy to maintain.
- Portable. Compared to other light field camera arrays, our system is very compact and easy to port for different applications.
- Affordable. By eliminating multiple workstations, network devices and external camera synchronization units, our system has a total cost under $\$10,000$.
- Adjustable. Since all cameras are mounted on a reconfigurable rig, we can easily adjust the camera baseline to achieve optimal reconstruction results.

Data Streaming. Streaming and storing image data from 9 cameras to a single single workstation is another challenge. In our system, each camera captures 8-bit images of resolution 1024x768 at 30fps. This indicates that we need to stream about 2Gbps data. To store the data, previous solutions either use complex computer farm with fast ethernet connections or apply compression on the raw imagery data to reduce the amount of data. For fluid surface acquisition, the use of compression scheme is highly undesirable as it may destroy features in the images. We therefore stream and store uncompressed imagery data. To do so, we connect an external SATA disk array to a data server (Intel SC5400BASENA chassis) as the data storage device. The disk enclosure houses 8 high performance 1TB SATA II 3Gb/s hard drives, and connects to a x4 PCI Express SATA controller via two high-quality 6 foot long locking Multilane mini SAS cables. This SATA array is capable of writing data at 500MB/s when configured to RAID 0.

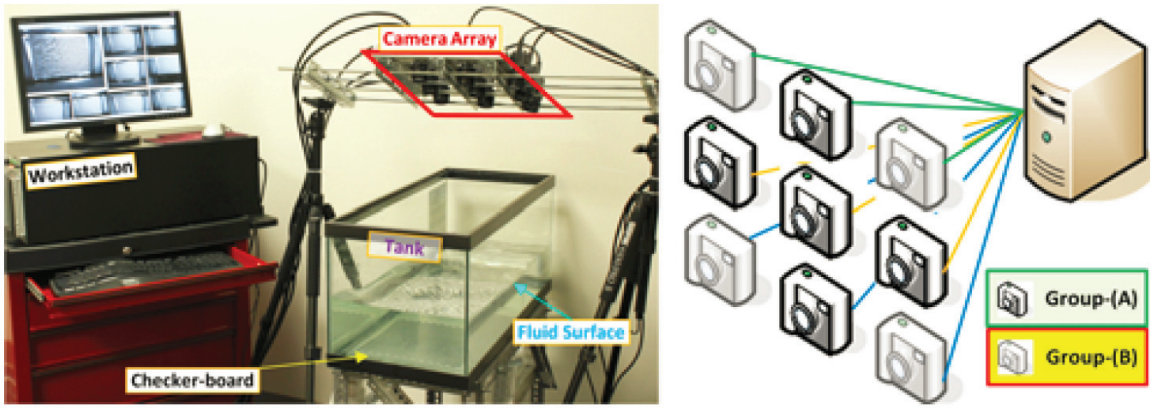


Figure 3.1: System setup for dynamic fluid surface acquisition. We divide the camera array into two groups (gray and black) and interleave the trigger for each group to double the frame rate.

Time-Divided Multiplexing. Since the Flea2 cameras can only achieve a maximum frame rate of 30fps, we have adopted a time divided multiplexing scheme to further improve the frame rate of our system. Our solution is similar to the Stanford light field high speed imaging scheme [128] that interleaves the exposure time at each camera. Specifically, we divide the camera array into two groups, four in one group and five in the other, as shown in the right of Fig.3.1. We set the exposure time of each camera to be 10ms to reduce motion blurs. While all cameras still capture at 30fps, we trigger the second camera group with a 1/60 second delay from the first one. We also develop special algorithms for warping the reconstruction result from the first group to the second so that our system is able to perform at 60fps. For details, please refer to [25].

Experiment Setup. We construct a plastic water tank of dimension $11in \times 17in \times 10in$. Compared with off-the-shelf glass water tanks, our plastic tank reduces inter-reflections and scattering so as to robustly track the feature points. We print a black-white checkerboard pattern on regular paper, laminate it, and then glue it to a planar plastic plate. We stick this plate onto bottom of the container and use it for both camera calibration and feature tracking.

Lens Specs. In our experiments, the choice of camera lenses is also crucial in our acquisition process. For example, a camera's field-of-view should be large enough to cover

the complete fluid surface. In our setup, we choose 45° wide angle lens with a focal distance of $12in$. Since all cameras are mounted on a reconfigurable rig, we can easily adjust the camera baseline to achieve optimal reconstructions.

Calibration. A number of options [73, 134] are available for calibrating the cameras in the array. Since the observable regions of our cameras have large overlaps, we directly use Zhang's algorithm [145] for calibration by reusing the checkerboard pattern mounted at the bottom of the tank. This approach also has the advantage of automatically calibrating all cameras under the same coordinate system and simplifies our feature warping scheme. We further perform color calibrations using the technique proposed by Ilie et al. [48].

Sample Results. To generate fluid motions without disturbing the camera setup, we use a hair dryer to blow air onto the surface. We start with reconstructing the first frame using cameras in group A, and we detect feature correspondences and apply specular carving to recover the normal field and then the height field. Fluid surface reconstruction results can be found in [25].

3.2 Hybrid Camera System.

Next we show how to extend the light field camera array to our proposed hybrid camera system. Capturing high quality color images under low light conditions is a challenging problem in computer vision. Images captured by commodity cameras are usually underexposed and very noisy. To reduce the sensor noise and hence improve the SNR, one can adopt two possible solutions: using a wide aperture or using a slow shutter. However, wide apertures will lead to shallow depth-of-field, i.e., only a small portion of the scene would be clearly focused; slow shutters, on the other hand, will lead to severe motion blurs in the presence of fast moving objects. It is also worth noting that using larger sensors will also help reduce the noise as the noise amplitude is approximately inverse-proportional to sensor size. For example, high-end digital SLRs with larger sensors perform much better than consumer digital cameras in low light imaging even with the same aperture setting.

From the spectrum perspective, one way to gather more photons is to capture Near Infrared (NIR) lights in addition to visible light, e.g., by using NIR sensitive cameras. A

downside is that NIR lights would lead to a predominance of red in color cameras. As a result, one needs to use an elaborately designed, camera-specific white balancing process for correcting the color. In fact, nearly all manufacturers equip their color cameras and camcorders[1] with an Infrared (IR) Cut Filter or ICF that behaves like the 486 optical filter[2] to block NIR light in color cameras as shown in Fig.3.3(c). An effective way to capture NIR lights without degrading image quality hence is to use monochrome cameras.

3.2.1 System Setup

Our goal is to combine the benefits of different types of cameras (with respect to aperture, shutter, resolution, and spectrum) by constructing a hybrid camera array. Fig.3.2 shows our proposed hybrid camera system: our prototype consists of two Pointgrey Grasshopper high speed monochrome (HS-M) cameras (top), two Pointgrey Flea2 high resolution monochrome (HR-M) cameras (bottom), and one single Flea2 high resolution color (HR-C) camera (center). All cameras are equipped with the same Rainbow 16mm C-mount F1.4 lens. We mount the five cameras on a T-slotted aluminum grid, which is then mounted on two conventional tripods for indoor applications. To deal with the long working range of outdoor applications, we also build a giant "tripod "from a 6 foot step ladder to hold the camera array grid, as shown in Fig.3.2(c). Similar to the camera system for dynamic fluid surface acquisition as discussed in Sec.3.1, we use the same workstation along with data streaming systems to acquire and store the images. We also apply similar techniques for camera synchronization and calibration.

Next, we explain our design principle. In our hybrid camera system, we use the HR-M camera with a large aperture to capture low-noise images. However due to the use of large apertures, the resulting images would have shallow depth-of-fields, i.e., strong defocus blurs for out-of-focus regions. We hence use two HR-M cameras focusing at different parts of the scene with the aim to fuse the focused regions in the two cameras. To handle fast motions,

[1] The only few exceptions such as Sony H9 have a NightShot model that can temporarily switch off the ICF to capture IR images.

[2] Courtesy of The Image Source©

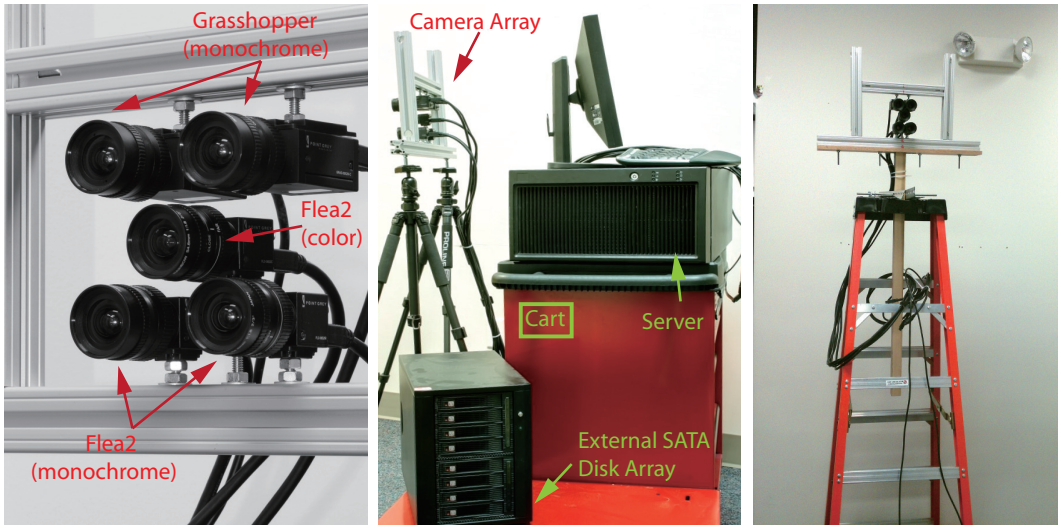


Figure 3.2: Our hybrid camera system for low-light imaging. (a) shows that our hybrid camera array consists of five cameras, two monochrome grasshopper (top), one color Flea2 (center), and two monochrome Flea2 (bottom). (b) shows the complete hybrid camera system for indoor applications, while (c) demonstrates our camera setup for outdoor low-light applications.

we use two HS-M cameras to capture the motion blur free images. However these images are usually dark, noisy and low contrast due to fast shutters. Color is very vital to many low light applications, therefore we use long exposure for the HR-C camera to capture reliable color information of the scene. The key disadvantage of using long exposure is that this could cause severe motion blurs for moving objects. In this dissertation, we propose new algorithms to fuse these imagery data to reconstruct high quality color images for low light applications.

In our setup, we choose to use monochrome cameras as the high speed cameras and the high resolution large aperture cameras, since they can gather more lights (visible and NIR) than color cameras. This is due to the fact that monochrome image sensors do not have an ICF or a "Bayer array"[3]. Bayer array can discard approximately 2/3 of the incoming light at every pixel, which is highly undesirable for low light imaging. To further reduce

[3] Virtually all color image sensors use the Bayer array pattern, except for the Foveon sensors used in the Sigma SD9/SD10 which captures all three colors at each pixel location.

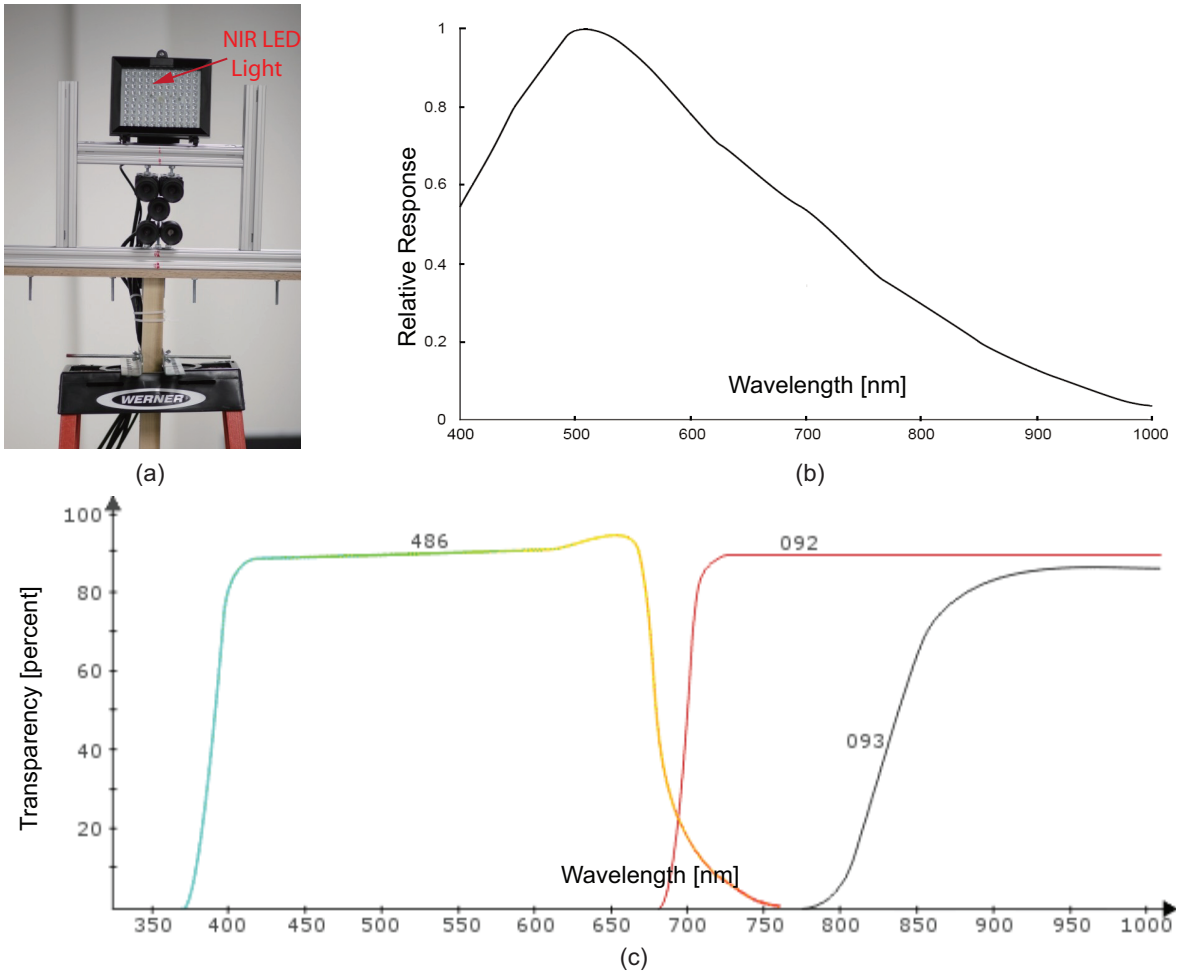


Figure 3.3: Hybrid camera setup for extreme low-light conditions. (a) shows our hybrid camera setup for extreme low-light conditions. An typical NIR illuminator with 98 LEDs is mounted on top of the camera rig. (b) shows the spectral sensitivity curve of Sony© CCD ICX204AL, used in the Flea2 monochrome cameras. (c) demonstrates the spectral characteristics graph of The Image Source© optical filters.

image sensor noise, the two HR-M cameras use a wide aperture and focus at two different depth layers. The two HS-M cameras have pixel size of 7.4 $\times$ 7.4 μm and work at 640 $\times$ 480 $\times$ 8bit@120fps whereas the two HR-M cameras have pixel size of 4.65 $\times$ 4.65 μm and perform at 1024 $\times$ 768 $\times$ 8bit@30fps. The last Flea2 color camera captures color images of size 1024×768 at 7.5fps. To reduce motion blurs, we configure the exposure time of the two HR-M cameras to be the same as the two HS-M cameras.

In theory, it is possible to use just one HS-M camera to capture fast motions. However, HS-M stereo cameras would make the warping of point spread function (PSF) onto the HR-C camera much more accurate than just one HS-M camera. This is because the disparity map estimated from one HS-M and two HR-M images would have lots of errors due to the difference in noise and defocus blur levels between them and the strong parallax existed in both directions on the image plane. Besides, the use of two HS-M cameras could also give us the strong extensibility to this hybrid camera system, for instance, high speed stereo video super-resolution, 3D TV, and etc.

To reduce motion blurs for HR-M cameras, we set their exposure time to be the same as the HS-M cameras. One major drawback of this dual HR-M camera setup is that we cannot get the low-noise infocus imagery for the entire depth range of the scene. Instead of adding more HR-M cameras focusing at the intermediate depth ranges, we carefully set the focuses of the two HR-M cameras. To fully utilize the current setup, we let the first HR-M camera focus at $H/2$, and the other one at $H/3$, where H is the hyperfocal distance. This would give us the optimal DOF from $H/4$ to infinity.

Under extremely low-light conditions, there would not be enough visible or NIR light even for our HR-M cameras. In this cases, similar to existing nightvision surveillance cameras, we install some active NIR LEDs around our camera system to improve the lighting condition. As shown in Fig.3.3(a), we mount on top of the camera rig an NIR illuminator with 300ft distance range and with wavelength of 850nm from YY Trade Inc. This NIR illuminator is compatible with our design, since the CCD sensor used in Flea2 monochrome cameras is capable to capture NIR of wavelength 850nm, as shown in Fig.3.3(b). Fig.3.4

Figure 3.4: A pair of sample images captured under extreme low-light conditions without/with NIR illuminator respectively.

shows a pair of sample images captured at a construction site with or without the NIR illuminator. We can see that this NIR illuminator is able to greatly improve the image quality under extremely low lighting conditions.

In summary, the HR-M cameras in our system aim to capture low-noise images but with strong defocus blurs, the HS-M cameras capture fast motions without motion blurs but their images are usually very noisy, and the HR-C camera provides reliable color information of the scene using slow shutters at the cost of strong motion blurs for the fast moving objects. In Chapters 4-7, we develop a class of computational techniques for combining the imagery data from our system for reconstructing high resolution, noise free, and motion blur free color video streams under low light conditions.

Chapter 4

MULTI-FOCUS FUSION

In this chapter, we present a novel multi-focus fusion technique. It uses a pair of images captured from different viewpoints, and at different focuses but with identical wide aperture size, called dual focus stereo pair (DFSP), as shown in Fig. 4.1. Wide apertures allow more light to be admitted to the camera and are suitable for low-light and fast motion imaging. However, they also lead to shallow depth-of-field (DOF) where pixels are sharp around what the lens is focusing on and blurred elsewhere. Hence, each image in an DFSP exhibits different defocus blurs and the two images form a defocus stereo pair, as shown in Fig. 4.2.

To model defocus blur, we introduce a defocus kernel map (DKM) that computes the size of the blur disk at each pixel. We derive a novel *disparity defocus constraint* for computing the DKM in DFSP, and integrate DKM estimation with disparity map estimation to simultaneously recover both maps. We show that the recovered DKMs provide useful guidance for segmenting the in-focus regions and multi-focus fusion.

4.1 Defocus Kernel Map

Similar to [31, 56, 66, 86, 92, 125, 131], we use Gaussian PSFs to effectively approximate defocus blurs. Recent papers on coded apertures [69] have shown that other types of PSFs may be more suitable for reducing blur kernels. We, however, choose to use the Gaussian PSF for its simplicity in modeling the DKM. In fact, we derive the DKM Constraint to directly correlate scene depth with Gaussian kernel sizes. In this paper, the defocus blur at every pixel p is modeled as:

$$I(p) = I_0(p) \otimes b(d(p), c) \tag{4.1}$$

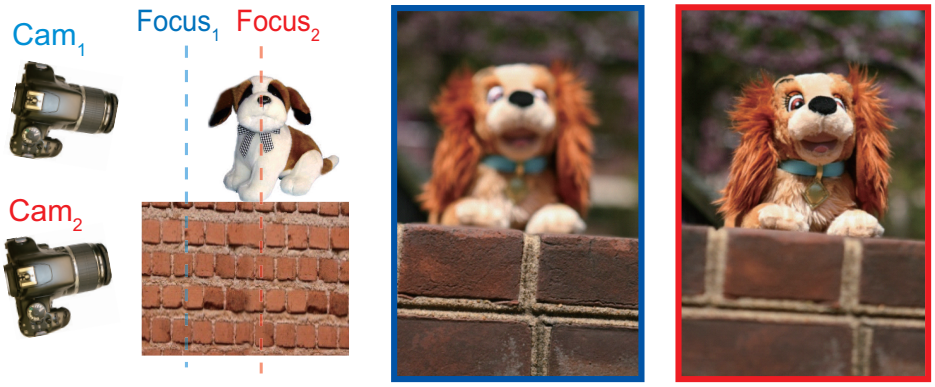


Figure 4.1: An dual focus stereo pair.

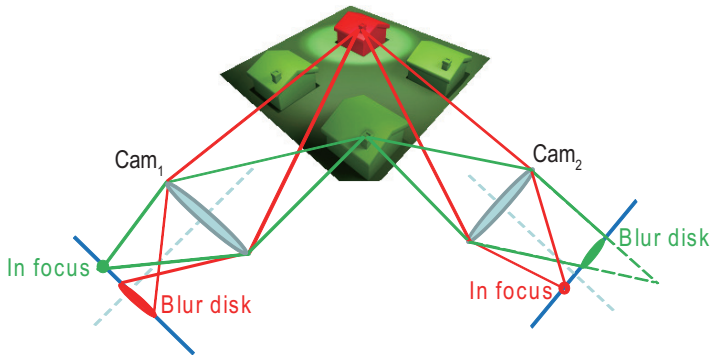


Figure 4.2: Each image in an DFSP focuses at different scene depths. A 3D point focused in one image will appear defocused in the other.

where $\otimes$ is the convolution operation. $I(p)$ is p's intensity after the defocus blur. $I_0(p)$ is p's intensity in the all-focus image. $b(d(p), c)$ is the blur kernel at p and is a function of p's depth $d(p)$ and the camera parameters c (i.e., camera aperture size and focal length). In this paper, we call b the Defocus Kernel Map or DKM.

In DFSP, since we only vary the scene focus while fixing the aperture size and the focal length, c will remain constant and hence we can use $b(p)$ to represent the blur disk size at every pixel p. We first derive $b(p)$ in terms of c and $d(p)$. Assume the camera uses a thin lens of focal length f, aperture size D, and f-number $N = f/D$, and its image plane is

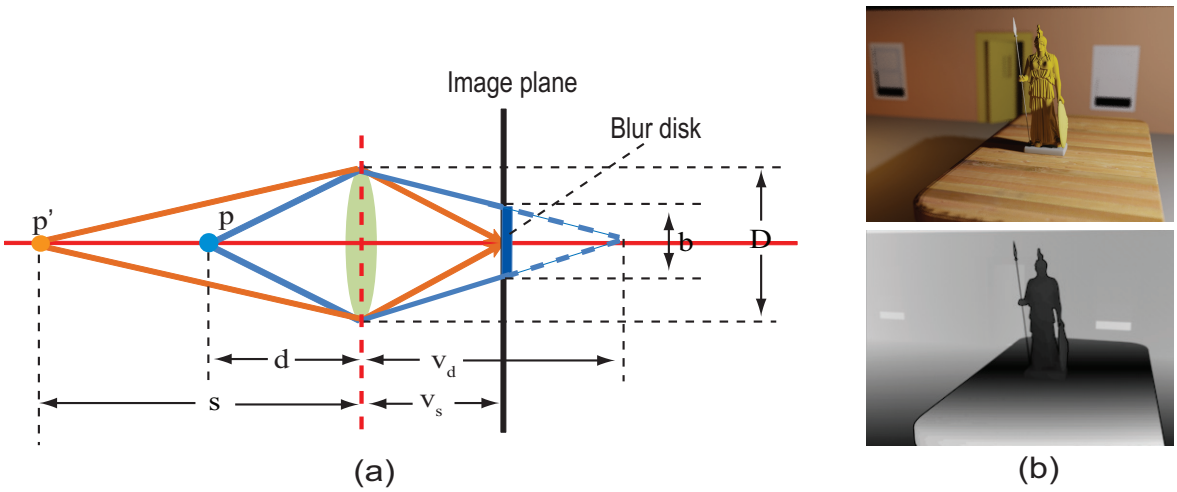


Figure 4.3: Defocus kernel map (DKM): (a) The blur disk diameter b depends on the aperture size and the depth of the scene point. (b) shows a sample DKM of a synthetic scene.

positioned at v_s away from the lens to focus objects at depth s from the lens. s and v_s then satisfy the thin lens equation:

$$\frac{1}{s} + \frac{1}{v_s} = \frac{1}{f} \tag{4.2}$$

Consider an arbitrary scene point p lying at depth $d(p)$, its image will focus at $v(p) = f \cdot d(p)/(d(p) - f)$ by the thin lens equation. If $d(p) < s$, then p's image will lie behind the image plane shown in Fig. 4.3(a). Thus, p will cast a blur disk of diameter $b(p)$, where

$$b(p) = \frac{(v(p) - v_s) \cdot D}{v(p)} \tag{4.3}$$

Substituting $v(p)$ and v_s with $d(p)$ and s using the thin lens equation and we have:

$$b(p) = \alpha \frac{| \, d(p) - s \, |}{d \cdot (s - \beta)} \tag{4.4}$$

where $\alpha = f^2/N$, and $\beta = f$. Eqn. 4.4 computes the blur disk size at every pixel in terms of its scene depth and we call it the *defocus constraint*.

4.1.1 Disparity Defocus Constraint

Next, we derive the *defocus constraint* in terms of the disparity map[1] in DFSP. DFSP is a pair of images I_1 and I_2 captured with the same f-number and focal length but focused at different scene depths. For every pixel p in I_1, its disparity $\gamma(p)$ (with respective to I_2) can be computed from its depth $d(p)$ as $\gamma(p) = K/d(p)$, where K is the multiplication of the baseline of the dual camera and their focal length, therefore it is constant for all pixels. Similarly, we can map the in-focus scene depth s in I_1 to its disparity γ_s. Substituting $d(p)$ and s with $\gamma(p)$ and γ_s in the *defocus constraint* Eqn. 4.4, we have:

$$b(p) = \tilde{\alpha}\frac{\mid \gamma_s - \gamma(p) \mid}{\tilde{\beta} - \gamma_s} \tag{4.5}$$

where $\tilde{\alpha} = \alpha/\beta$ and $\tilde{\beta} = K/\beta$. We call Eqn. 4.5 the *disparity defocus constraint*. Similarly, we can compute the DKM b of I_2 with respect to I_1. For the rest of the paper, we, by default, refer the DKM to the one associated with I_1.

4.2 Defocused Stereo Matching

In this section, we show how to simultaneously recover the DKMs and the disparity map. We assume that each image in an DFSP focuses at some scene objects/features. This is a common practice in photography, especially when one uses a hand-held camera. Our algorithm starts with finding SIFT feature correspondences [78] between the DFSP and apply [88] to rectify the images. Next, we extract the salient features in each rectified image and estimate their initial disparity values to recover the camera parameters $\tilde{\alpha}$ and $\tilde{\beta}$ (Sec. 4.2.1). We then integrate the defocus kernel map estimation with the disparity map solution process using the *pair-wise defocus constraint*. Finally, we iteratively refine the camera parameters, the DKMs, and the disparity map. Fig. 4.4 illustrates the processing pipeline of our algorithm.

---

[1] The disparity map here refers to the one computed between the corresponding all-focus images.

4.2.1 Recovering Camera Parameters

We first develop a simple but effective algorithm to recover the camera parameters. Since we have two unknowns $\tilde{\alpha}$ and $\tilde{\beta}$, we use two disparity defocus constraints to solve for them. To do so, we estimate the disparity defocus constraints for the in-focus pixels Ω_1 and Ω_2 in I_1 and I_2, respectively. Assuming Ω_1 has disparity γ_1 and casts a blur disk of size b_2 in I_2, and Ω_2 has disparity γ_2 and casts a blur disk of size b_1 in I_1, the two *disparity defocus constraints* are:

$$\begin{cases} b_1 = \tilde{\alpha}\dfrac{\mid \gamma_1 - \gamma_2 \mid}{\tilde{\beta} - \gamma_1} \\ b_2 = \tilde{\alpha}\dfrac{\mid \gamma_2 - \gamma_1 \mid}{\tilde{\beta} - \gamma_2} \end{cases} \tag{4.6}$$

Since an DFSP focus at different scene depths, $\gamma_1 \neq -\gamma_2$ and Eqn. 4.6 are non-degenerate. Our goal is to find γ_1, γ_2, b_1, and b_2, and solve for $\tilde{\alpha}$ and $\tilde{\beta}$.

To find γ_1 and γ_2, we first compute the salient features by applying a high-pass filter on I_1 and I_2. To minimize outliers, we blur each image I_i using a small Gaussian kernel and then subtract the blurred image from I_i. We use Gaussian kernels as they are coherent with the defocus blur model and effectively suppress aliasing artifacts such as ringing. We then threshold the high-pass filtered images to obtain initial salient maps, as shown in Fig. 4.5(a) and (b). We also use the graph-cut algorithm to compute the initial disparity map and assign the computed disparity value to all salient feature points. We assume all salient features in their corresponding images correspond to the same depth and, hence, have the same disparity. To remove outliers, we use the median disparity value of feature points in I_1 and I_2 as γ_1 and γ_2.

To find b_1 and b_2, we locate the corresponding pixels and exhaust all possible kernel sizes b. To find b_2, for each feature point p in the salient map of I_1, we apply Gaussian blur of size b_2 at pixel p and compute the difference between p and $q = p + \gamma_1$ in I_2. We find the optimal b_2 that minimize the summed squared difference for all salient features in I_1.

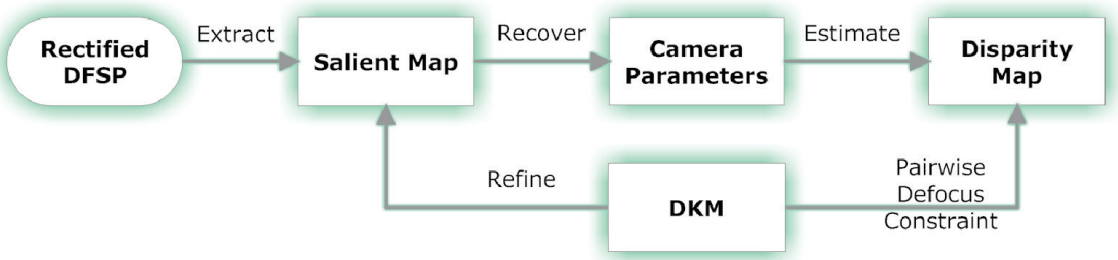


Figure 4.4: The processing pipeline of multi-focus fusion technique.

Similarly, we obtain the optimal b_1. Finally, we combine b_1 and b_2 with γ_1 and γ_2 to solve for $\tilde{\alpha}$ and $\tilde{\beta}$ using disparity defocus constrain Eqn. 4.5.

4.2.2 DKM-Disparity Markov Network

Classical stereo matching methods model the disparity map as a Markov Random Field (MRF). The problem can be treated as assign a label $\gamma : P \rightarrow \Gamma$ where P is the set of all pixels and Γ is a discrete set of labels corresponding to different disparities. Graph-cut [12, 59, 60] and Belief Propagation [33, 109, 110] can be used to find the optimal labeling. Since the DKMs can be directly derived from the disparity map (Eqn.5), they share similar MRF properties as the disparity map, i.e., they are smooth except crossing a boundary. Therefore, we can integrate DKM estimation with the graph-cut based disparity map estimation process.

We define the energy function E as:

$$E(\gamma) = \sum_{p \in I_1} E_r(p, \gamma(p)) + \sum_{p_1, p_2 \in N} E_s(\gamma(p_1), \gamma(p_2)) \tag{4.7}$$

where the data penalty term E_r describes how well the disparity γ fits the observation, the smoothness term E_s encodes the smoothness prior of Γ, and N represents the pixel neighborhood in Image I_1.

In our implementation, we use the similar smoothness term as in [60]. The data penalty $E_r(p, \gamma(p))$ measures the appearance consistency between pixel p in I_1 and pixel $q = p + \gamma(p)$ in I_2. Recall that I_1 and I_2 have different focuses. Thus, even with the correct

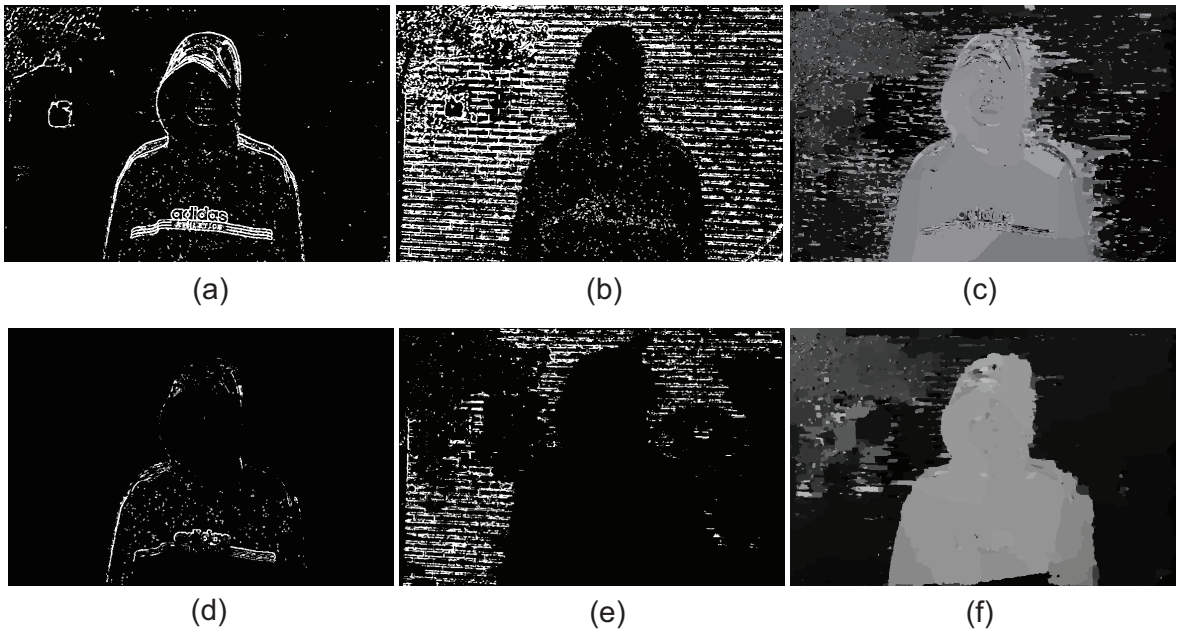


Figure 4.5: The recovered disparity map. (a) and (b) show the initial salient feature maps of the DFSP in Fig.7. (c) shows the initial disparity map. (d) and (e) show the refined salient feature map after pruning the outliers. (f) shows the recovered disparity map after 2 iterations.

disparity γ, the appearance of p and q may appear significantly different due to defocusing. Therefore, we cannot directly compare the intensity between $I_1(p)$ and $I_2(q)$.

Note that given γ and the recovered camera parameters $\tilde{\alpha}$ and $\tilde{\beta}$, we can directly compute the defocus blur kernel b_p and b_q at pixel p and q using Eqn. 4.5. Assuming p is less blurry than q, we can apply additional Gaussian blur G_σ to p in I_1 and then compared the blurred result with q. We call the resulting images I_1^* and I_2^* an *equally-defocused* pair:

$$
\begin{aligned}
I_1^*(p) &= \begin{cases} I_1(p) \otimes G_\sigma, & b_p < b_q \\ I_1(p), & otherwise. \end{cases} \\
I_2^*(q) &= \begin{cases} I_2(q) \otimes G_\sigma, & b_q < b_p \\ I_2(q), & otherwise. \end{cases}
\end{aligned} \tag{4.8}
$$

$$
\sigma = \sqrt{\mid b_p^2 - b_q^2 \mid}
$$

Finally, we define E_r as

$$E_r(p, \gamma(p)) = \min\{0, (I_1^*(p) - I_2^*(p + \gamma(p)))^2 - K_2\} \tag{4.9}$$

where the truncation threshold K_2 is used to reduce noise and remove outliers. We use graph-cut to solve for the optimal disparity map and apply the disparity defocus constraint for computing the DKMs. In theory, we could further improve the disparity map by modeling the occlusion boundaries [60, 109]. In practice, we find it sufficient and robust to use the estimated DKM for segmentation in the following sections.

Once we obtain the disparity map and the DKMs, we can refine the salient feature maps by removing points whose disparity deviates from the median disparity of all feature points. We then re-estimate the camera parameters (Sec. 4.2.1). We repeat this iterative refinement 2 to 3 times. Fig. 4.5(f) shows the final estimated disparity map.

4.2.3 DKM-based Segmentation

The recovered DKMs can be used to robustly segment the in-focus region in each image of DFSP. Specifically, we treat the segmentation problem as a labeling problem on the DKM and use only two labels, S for the foreground and T for the background. To find the optimal labeling, we define the energy function E as:

$$E(L) = \lambda \cdot \sum_{p \in P} E_r(p, L(p)) + \sum_{p,q \in N} E_b(p, q, L(p), L(q)) \tag{4.10}$$

where P represent all pixels in the image, N represents the pixel neighborhood, $L(p)$ is the labeling at p. The nonnegative coefficient λ specifies a relative importance of region penalty E_r and boundary penalty E_b. To model discontinuities in labeling, we model E_b as,

$$E_b(p, q, L(p), L(q)) = \begin{cases} \dfrac{\exp(-(L(p) - L(q))^2)}{\| p - q \|}, & L(p) \neq L(q) \\ 0, & otherwise. \end{cases} \tag{4.11}$$

Unlike previous segmentation methods [13,99] that define the region penalty E_r using the histogram distribution or Gaussian mixture models from user-specified foreground/ground samples, we directly compute E_r in terms of the defocus kernel map B. Notice that a smaller

(a) (b) (c)

Figure 4.6: Problems with low-light imaging. (a) An image captured with a small aperture and fast shutter appears underexposed. (b) shows the histogram stretched result of (a). (c) An Image captured using a small aperture and slow shutter exhibits severe motion blurs.

defocus kernel b corresponds to a higher likelihood that the pixel is in-focus. Thus, we simplify E_r as

$$E_r(p, S) = \begin{cases} K_3 \cdot (M - b_p), & b_p < M \\ 0, & otherwise \end{cases} \tag{4.12}$$

$$E_r(p, T) = \begin{cases} b_p - M, & b_p \geqslant M \\ 0, & otherwise \end{cases} \tag{4.13}$$

where M is the maximum size of circle-of-confusion that would be considered in focus. K_3 is a positive scaling factor for balancing the region penalty between foreground and background. In our experiments, we set $M = 5$ and $K_3 = (b_{max} - M)/M$, where b_{max} corresponds to the maximum size of the blur disk. Fig. 4.7(d) shows a sample segmentation result in Sec. 4.3.1.

4.3 Applications

In this section, we demonstrate how to apply multi-focus fusion technique for low-light imaging, multi-focus photomontage and authomatic defocus matting.

4.3.1 Low Light Imaging

Capturing high quality images under low light is a challenging problem. Images captured with a regular aperture and shutter setting are commonly underexposed and noisy,

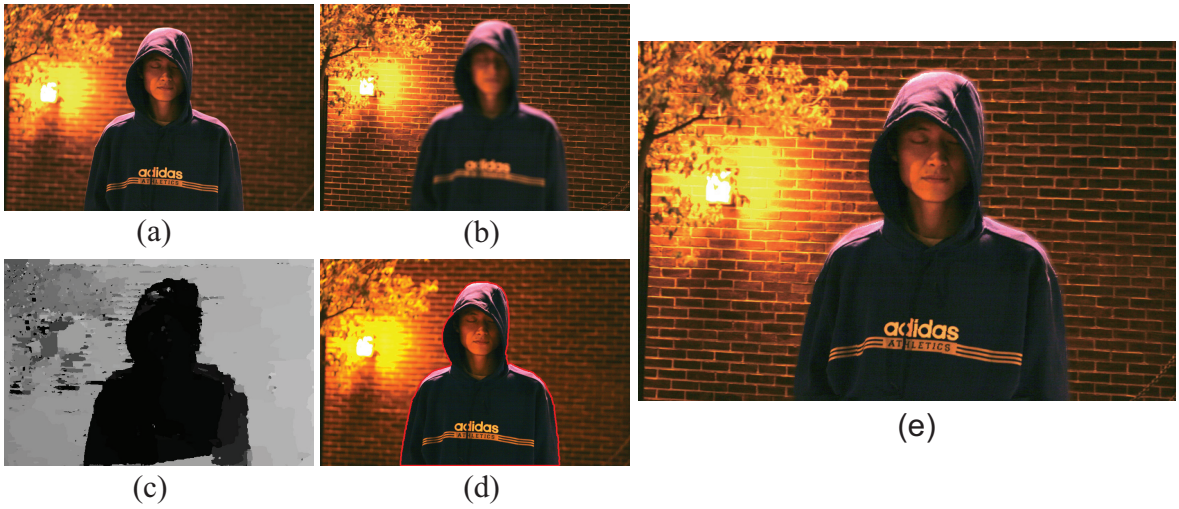


Figure 4.7: DFSP for low-light imaging. (a) and (b) show an DFSP. We use the DKM (c) to segment the in-focus regions in the first image (d). Finally, we warp the segmented in-focus regions to the second image using the estimated disparity map to form a nearly all-in-focus image (e). (See Appendix A for the permission letter to use these pictures.)

as shown in Fig. 4.6(a). One possible solution is to use slow shutters. However, slow shutters can cause significant image blur due to scene and/or camera motions. In DFSP, slow shutters can be particularly problematic as we use a hand-held camera to capture the images, as shown in Fig. 4.6(c). The resulting motion blur is difficult to correct as they are not spatially-invariant [34, 102]. Another possible solution is to denoise the images, e.g. via principal component / tensor analysis [144]. However, a large number of images (~ 20) are often required for robust denosing. Alternatively, we can use wide apertures in place of slow shutters, which would lead to shallow DOFs and only part of the scene can be clearly focused, as shown in Fig. 4.7(a) and (b).

In this section, we use DFSP to enhance low-light imaging. We place two synchronized digital SLR cameras adjacent to each other. The two cameras use the identical aperture size, with one focusing at the foreground and the other at the background. Instead of deblurring the images, we explore effective fusion methods to combine in-focus regions in both images of DFSP.

Recall that we use the recovered DKMs to segment the in-focus region (Sec. 4.2.3). A

naive approach is to directly warp the segmented region using the disparity map. In practice, the estimated disparity map can still be noisy and discontinuous along the segmentation boundaries. Hence, directly warping the boundary will cause jig-jagged or scattered edges in the final fused image. To resolve this issue, we develop a *Snake*-based [55] contour warping algorithm.

Given the boundary V_1 of the in-focus region in I_1, our goal is to find its optimal *target* boundary V_2 in I_2 so that when V_1 is warped to V_2 using the recovered disparity map, it will be continuous with respect to the background in I_2. To do so, we define the energy function for all pixels $\tilde{p}$ on contour V_2 as:

$$E_{contour} = \sum_{all \tilde{p}, \tilde{p} \in V_2} (E_{image}(\tilde{p}) + \zeta \cdot E_{shape}(\tilde{p})) \tag{4.14}$$

where E_{image} describes appearance similarity and E_{shape} describes the shape similarity.

Recall that we cannot directly compare the intensity between pixel p on V_1 and $\tilde{p}$ on V_2 since they have different defocus levels. Therefore, we first use the recovered DKMs to compute the equally-defocused image pair I_1^* and I_2^* using Eqn. 4.8 . We then measure the similarity between p and $\tilde{p}$ using I_1^* and I_2^* as:

$$E_{image}(\tilde{p}) = \mid F(I_1^*(p)) - F(I_2^*(\tilde{p})) \mid \tag{4.15}$$

where $F(I_1^*(p)) = (I_1^* \otimes G)(p)$, $F(I_2^*(\tilde{p})) = (I_2^* \otimes G)(\tilde{p})$ and G is a Gaussian kernel that serves as a weighting function.

In addition, we enforce the shape similarity between the two contours. Specifically, we measure the similarity of the first and second order differential geometry attributes between the corresponding point p and $\tilde{p}$ on V_1 and V_2 as:

$$E_{shape}(\tilde{p}) = \parallel V_1'(p) - V_2'(\tilde{p}) \parallel + \parallel V_1''(p) - V_2''(\tilde{p}) \parallel \tag{4.16}$$

and we apply a Greedy algorithm similar to *Snakes* [55] to find the optimal boundary in V_2.

The computed disparity inside the segmented region can also be noisy. Therefore, we only use the disparity estimation on the boundary pixels and smoothly interpolate the disparity value for the interior pixels. Finally, we use the interpolated disparity map for

warping all pixels inside the in-focus region to the final fused image. Fig. 4.7(e) shows such an example.

4.3.2 Multi-focus Photomontage

Finally, we apply DFSP for creating multi-focus photomontage which virtually focuses at multiple scene depths. Synthesizing multi-focus photomontage can benefit many applications. For example, in confocal microscopy, only tissues lying near the scanning layer can be clearly imaged and it is highly desirable to combine the in-focus regions from all layers. Another interesting application is to synthesize novel defocusing effects, e.g., by focusing at both the foreground and the background while defocusing at the in-between ground.

Our DFSP-based multi-focus photomontage differs from existing multi-focus fusion approaches [1, 45] in several ways. First, we use images captured from different viewpoints whereas existing approaches assume that the images are captured from or can be warped to the same viewpoint. Second, most photomontage techniques require using a large number of images to accurately identify the in-focus regions while we only use a pair of images. Finally, unlike digital photomontage [1] that relies on user inputs, our DFSP-based photomontage method is fully automatic.

Given an DFSP, we start by segmenting the in-focus region using the recovered DKMs. In low-light imaging (Sec. 4.3.1), we directly warp the segmented in-focus region from one image to the other. We assume that scene objects lie on different depth layers and rely on Snake for locating the occlusion boundary. For multi-focus photomontage, scene depth may vary smoothly (e.g., the mulch ground in Fig. 4.8). Therefore, directly warping the foreground region will lead to sudden changes of blurriness across the warping boundary as shown in Fig. 4.8(g).

To resolve this issue, we compute the DKM for each image in the DFSP to determine the defocus blur kernel size near the warped boundaries. We then blur the warped in-focus boundary region accordingly to maintain smooth transitions. Specifically, given the warped boundary, we first erode and dilate the boundary Ω by a fixed width to form a band. Our goal

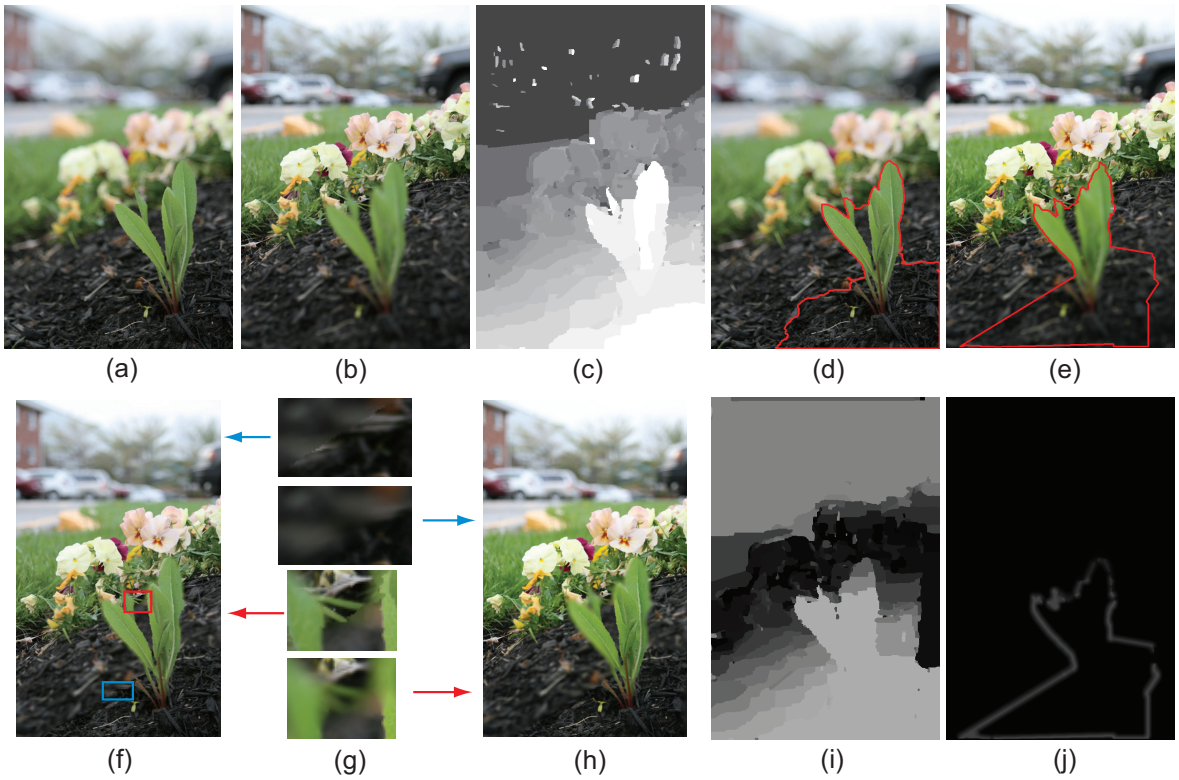


Figure 4.8: Multi-focus photomontage on a continuous scene. (a) and (b) show an DFSP of a garden scene. The first image focuses at the foreground sprouts and the second at the middle-ground flower. (c) and (d) are the recovered disparity map and the segmentation result. (e) shows the warped boundary using snake-based algorithm. (f) shows the fusion result by directly warping the segmented foreground to the second image. Notice the blurriness changes abruptly across the boundary (g). (h) maintains smooth boundary by blurring the segmentation boundary using the adaptive kernel map (j) and the DKM (i).

is to determine how much blur should be applied to each pixel inside the band after warping. We call this process adaptive boundary blurring.

To maintain the same sharpness/blurriness outside the band, we set the adaptive blur kernel size to be zero on both the interior and exterior boundaries Ω^+ and Ω^- that are obtained by erosion and dilation. We determine the adaptive blur kernels on Ω from the estimated DKM and use natural neighbor coordinates [105] to interpolate the missing pixels inside the band. Fig. 4.8 (e) and (j) show the warped boundary and the adaptive blur kernel map. Fig. 4.8 (f) and (h) compares the results between direct warping and adaptive blurring.

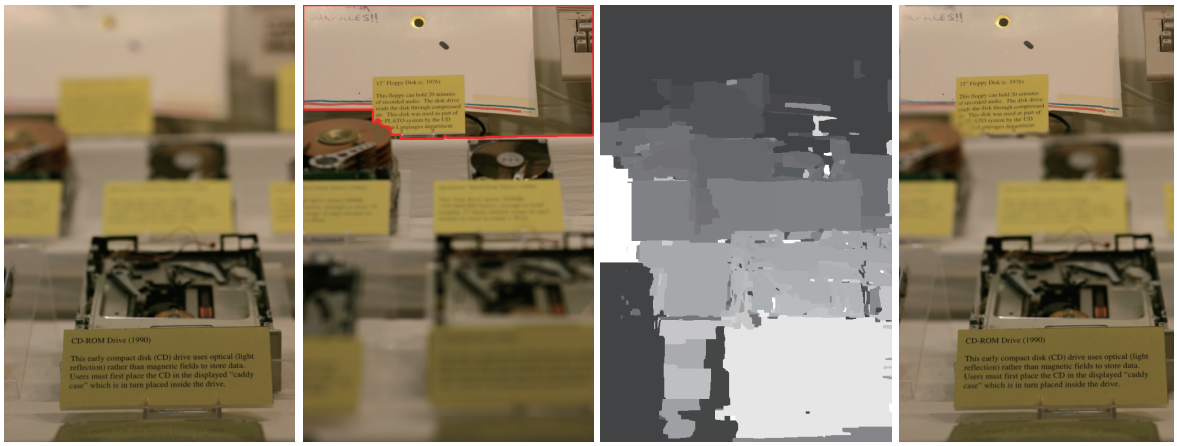


Figure 4.9: Multi-focus photomontage on a computer museum scene. The first image (a) in the DFSP focuses at the foreground tags and second (b) at the background tags. (c) shows the estimated disparity map . The segmented background boundary is shown as red in (b). (d) shows the final fused image with a novel multi-focus effect where only the middle ground is defocused.

Using adaptive blurring, our photomontage technique is able to maintain smooth transitions from the in-focus region to the out-of-focus region.

4.3.3 Other Applications: Automatic Defocus Matting

Next, we apply DFSI technique for automatically extracting the alpha matte. The problem of image matting has been studied for decades. Recently, several multi-image based matting methods have been proposed to automatically extract the matte. For example, defocus video matting [81] uses a special imaging system to capture multiple images of the scene from the same viewpoint and with different focuses. It then automatically classifies the image into a foreground Ω_F, a background Ω_B, and an unknown region Ω_U by analyzing the defocus blur. However, multi-image based methods are sensitive to calibration errors, camera shake, and foreground motions.

In this paper, we use the recovered DKMs from the DFSI pair to automatically generate the trimap. For trimap-based matting schemes, the quality of the alpha matte relies heavily on the estimation accuracy of the unknown region Ω_U. One way to generate Ω_U is to

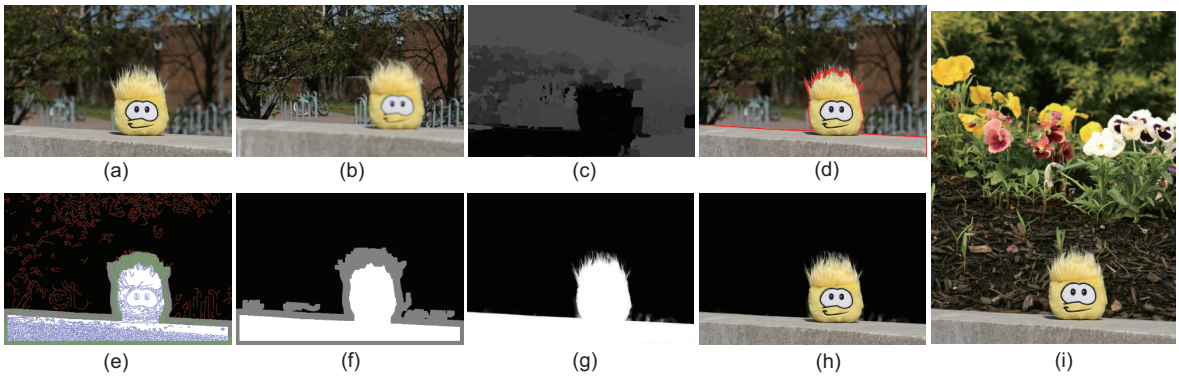


Figure 4.10: DFSP for Alpha Matting. (a) and (b) show a DFSP of the Disney club penguin scene, with the left image focusing at the penguin and the right at the background tree. Notice the strong parallax between the images. We recover the DKM (c) and use it to segment the foreground (d). We create a trimap (f) using morphologic operations based on edge heuristics (e). (g) and (h) show the recovered alpha matte and foreground using Robust Matting [122]. (h) We create a composite image using a new background.

erode the estimated foreground and background as:

$$\Omega_U = \overline{erode(\Omega_F, \tilde{b}) \cup erode(\Omega_B, \tilde{b})} \tag{4.17}$$

where $erode$ is a morphological operator [43] and $\tilde{b}$ is the width of erosion. For DFSI, we can compute Ω_F and Ω_B by the DKM-based segmentation and $\tilde{b}$ by averaging the blur kernel of all pixels in Ω_B.

Such trimap generation method works well when the segmented foreground boundary is relatively accurate. However, for DFSI, the boundary of the in-focus region is extracted based on the estimated disparity map. In presence of fuzzy boundaries, it is difficult to distinguish defocus blur from fuzziness and the resulting foreground boundary can be inaccurate as shown in Fig. 4.10(d).

We present a new scheme for generating the trimap by exploiting edge continuities. The key observation here is that, for fuzzy objects, strong edges along the foreground boundary should be considered as part of the unknown region. Therefore, we grow the unknown region along the edges directions. Specifically, we first use Canny edge detector [17] to

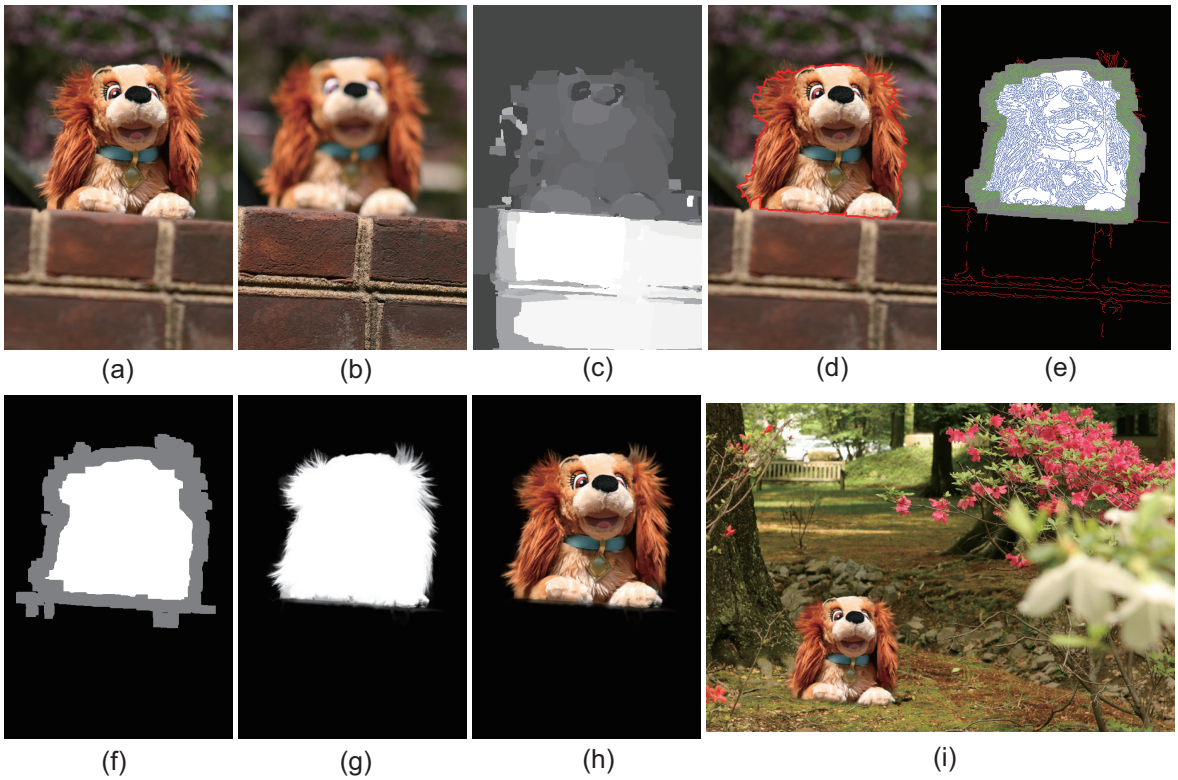


Figure 4.11: DFSP for Background Matting. (a) and (b) show a DFSP of a toy dog scene, with the first image focusing at the foreground bricks and the second at the middle ground toy dog. (c)-(f) show the recovered disparity map, segmented boundary, edge heuristic overlaid with trimap by Eqn. 4.17, and our trimap respectively. (g) and (h) show the extracted alpha matte and foreground. (i) shows the composite result.

locate the strong edges, then dilate these edges, and combine them with Ω_U to form the extended unknown region Ω_U^* as

$$\begin{aligned} \Omega_1 &= dilate(\Omega_U, 2\tilde{b}) \cap dilate(canny(I), \tilde{b}) \\ \Omega_U^* &= \Omega_1 \cup \Omega_U \\ \Omega_F^* &= \Omega_F \cap \overline{\Omega_U^*} \end{aligned} \tag{4.18}$$

Once we obtain the trimap, we use Robust Matting [122] to estimate the alpha matte, the foreground, and the background. Fig. 4.10 and 4.11 show the automatic defocus matting results using our approach.

4.4 Results

In our experiments, all images were captured using a Canon Rebel XTi digital SLR camera with Canon EF 24-70mm f/2.8L lens. We first recover the DKMs and the disparity map. We capture an DFSP with Av 2.8 and Tv 1/8 as shown in Fig. 4.7(a) and (b). We start with computing the initial salient feature maps (Fig. 4.5 (a) and (b)) and the disparity map (Fig. 4.5 (c)) for estimating the camera parameters. Fig. 4.5 (d) to (f) show the refined salient feature maps and the disparity map after 2 iterations. Our iterative optimization effectively removes outliers in salient feature maps and significantly improves the disparity map.

In Fig. 4.6, we demonstrate using multi-focus fusion technique for acquiring images under low-light. An image captured under a regular aperture and shutter setting (Av 11, Tv 1/8) appears underexposed as shown in Fig. 4.6(a). Gamma correction and histogram equalization can be used to enhance contrast. However, the resulting image is still noisy due to insufficient lighting (Fig. 4.6(b)). To increase exposures, we use a slower shutter (Tv 2) (Fig. 4.6(c)). However, the image contains severe motion blurs due to hand shakes. Next, we capture an DFSP (Fig. 4.7(a) and (b)). We recover the DKM (Fig. 4.7(c)) and use it to segment the in-focus region. Finally, we warp the segmentation result to Fig. 4.7(e). The final synthesized image preserves final details with minimal noise and motion blur.

In Fig. 4.8, we demonstrate our multi-focus photomontage technique. We capture an DFSP of a garden scene using F2.8. The first image focuses at the foreground sprouts and the second at the middle ground flower. Fig. 4.8(c) shows the recovered disparity map. We use the DKM to segment the foreground region as shown in red in Fig. 4.8(d). Next, we warp the foreground to the second image (Fig. 4.8(f)). However, directly warping the foreground region incurs abrupt changes of blurriness across the warping boundary. Therefore, we further compute an adaptive kernel map (Fig. 4.8(j)) from the foreground boundary. We use the second DKM (Fig. 4.8(i)) to adaptively blur the boundary. Fig. 4.8(g) shows the close-up views of the fusion results with and without adaptive blur.

In Fig. 4.9 (a) and (b) we create a multi-focus photomontage on a computer museum scene where the images focus at different tags. Fig. 4.9(b) shows the background segmentation result for the second view. We then warp the background to the first view and apply

adaptive blur to maintain smooth boundary transition. Fig. 4.8(d) illustrates a novel multi-focus photomontage effect by keeping both the foreground and background in-focus and the middle ground out-of-focus.

In Fig. 4.10, we demonstrate automatic defocus matting of a Disney club penguin scene using DFSI. Fig. 4.10(a) and (b) were captured with a wide aperture and fast shutter (Av 6.3, Tv 1/640). Fig. 4.10(a) focuses at the foreground Disney club penguin and Fig. 4.10(b) at the background trees. To extract the matte, we recover the DKM (Fig. 4.10(c)) and use it to segment the foreground (d). Since the right half of the stone bench appear textureless due to slight defocus blur, its DKM is not accurate. However, the DKM is only used to initialize the segmentation process, therefore we are still able to accurately segment the foreground region. Fig. 4.10(e) shows the edge heuristics for trimap computing: all edges are detected by the Canny operator and overlaid with the trimap estimated using Eqn. 4.17, where red, green, and blue edges lie in the estimated background, unknown, and foreground regions, respectively. Fig. 4.10(f) shows the final trimap created by the boundary growing algorithm (Sec. 4.3.3). Fig. 4.10(g) and (h) show the recovered alpha matte and foreground using robust matting. In Fig. 4.10(i), we create a composite image by using a new background. In Fig. 4.11, we show another example using DFSI defocus matting in a toy dog scene.

4.5 Discussions

We have demonstrated a novel multi-focus fusion technique. It captures a pair of images from different viewpoints and at different focuses but with identical wide aperture size. Table 4.1 summaries the major difference between our multi-focus fusion technique and other related work. Compared with coded aperture imaging [69], our method does not require using specially designed optical systems or modifying the camera. Furthermore, our method can be directly applied to enhance low-light imaging by coherently increasing the aperture size in both image, which is hard to achieve using coded apertures. Compared with color-filtered aperture imaging [5], our method avoids complex color calibration procedures and works robustly in presence of single-colored scene objects.

Table 4.1: Comparisons with state-of-the-art methods

	# of Images	Coaxial	Special Equipment	Output
McGuire [81]	3	Yes	Optical Bench + Beam Splitters	Matte
Joshi [54]	8	No	1x8 Camera Array	Matte
Levin [69]	1	No	Coded Aperture Masks	Depth Map + Matte
Bando [5]	1	No	Color Filtered Aperture	Depth Map + Matte
Harsinoff [45]	451 $\sim$ 793	Yes	None	Depth Map
Our method	2	No	None	Depth Map + Matte

Our multi-focus fusion technique, however, has several limitations. In our implementation, we have used spatially variant Gaussian PSFs to approximate defocus blur kernels. Since the kernels are no longer Gaussian near occlusion boundaries, we are unable to accurately estimate scene depth for the boundary pixels and our fusion technique generates visual artifacts such as bleeding and discontinuity. These artifacts have also been observed in the gathering methods [28, 66] used to render DoF effects in computer graphics. The scattering method [89], in theory, can help reduce the bleeding/distinuity artifacts. However, it requires highly accurate scene geometry to avoid aliasing. Since our technique only uses two images, our estimated scene depth map cannot reach the accuracy level.

Another major limitation of our approach is that we rely on the disparity map solution for recovering the DKMs. In our approach, we compute the disparity labeling by blurring the relatively sharp pixel with the optimal kernel to match the blurrier one. Levin et al. [69] have shown that due to the frequency characteristics of the circular aperture, multiple kernels exist that produce similar blurry results. Therefore, our disparity map may introduce large errors in regions that are defocused in both images. Note that, for our hybrid camera system, if the fused image still have noticeable sensor noise, we can further improve the fused image quality via the recently proposed defocus denoising method [103].

Chapter 5

MULTISPECTRAL DENOISING

In this chapter, we present a framework to remove sensor noise in low-light images captured by our hybrid camera system. We first preprocess the captured images from HS-M cameras using our exposure fusion technique. These images are usually very dark because HS-M cameras are set to use fast shutters to prevent object motion blurs. Next, we apply the dual focus stereo imaging technique presented in Chap.4 to generate all-in-focus image patches from HR-M image pairs and use the results for multispectral denoising. At last, we design an novel optimization scheme that uses image patches from the low noise HR-M images as the gradient prior for regularizing the data fidelity term of the objective function. The denoised results can be later used to estimate object motions for motion deblurring of HR-C images. We will discuss our hybrid camera design for motion deblurring in the next chapter.

5.1 Noise Model

We first introduce the noise model in low-light imaging. Digital cameras normally produce three common types of noise: random noise, fixed pattern noise, and banding noise. Fixed pattern noises are often referred to as "the hot pixels". They have the same appearance in images taken under the same settings (exposure, ISO speed, and temperature). We assume fixed pattern noise is repeatable and hence is easy to remove. In our setup, we assume that fixed pattern noise has been removed via the camera's in-chip preprocessing. Banding noise is introduced by the camera when it reads data from the digital sensor. Not all types of digital cameras would generate banding noise, although high ISO speeds, shadows or photo brightening may lead to such noise. In this chapter, we model banding noise and random noise together as white, zero-mean Gaussian noise, with a known standard deviation σ.

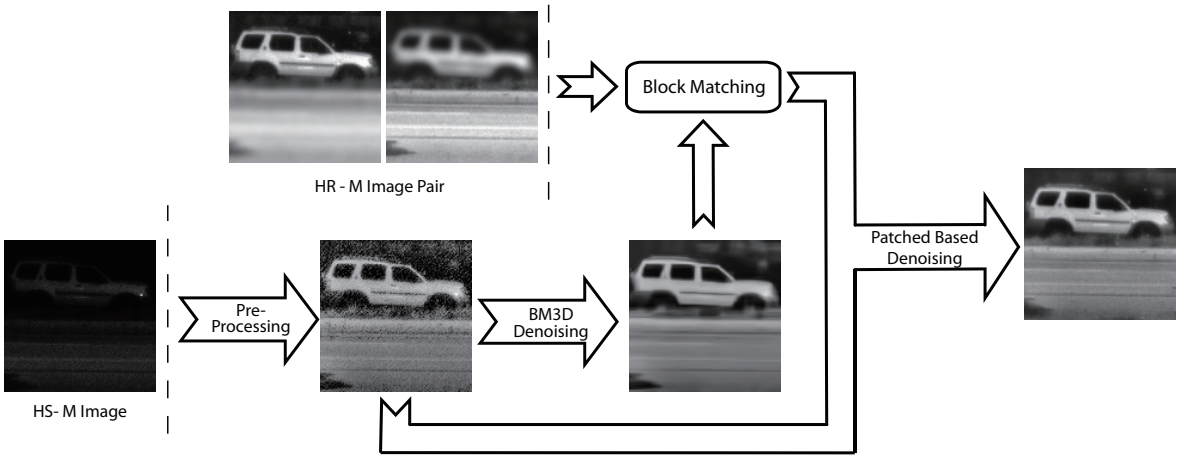


Figure 5.1: The processing pipeline of multispectral image denoising. We first preprocess the HS-M images to improve the brightness of dark regions, and then apply BM3D denoising algorithm to partially remove sensor noise. Next we find low-noise patch priors from HR-M images for multispectral denoising of HS-M images. Finally, we design an alternating minimization algorithm to denoise the preprocessed HS-M images.

Similar to [90], we assume the noisy image I can be decomposed into a latent image $\hat{I}$ and noise η as:

$$I(x) = \hat{I}(x) + \eta(x)\ , \quad \text{where } x \in \mathbf{X}\ , \eta \sim \mathcal{N}\{0, \sigma^2 I\}\ , \tag{5.1}$$

where x is the pixel index. In this chapter, we assume the latent and the noisy images are of size $M \times N$ pixels.

5.2 Outline of Our Approach

In our hybrid camera system as shown in Fig.3.2, for every color frame I_i^{hrc} captured by the HR-C camera, we have 4 synchronized high resolution grayscale frames $I_p^{hrm_j}$ ($p = 4i, \ldots, 4i+3$; $j = 0, 1$) captured by each of the two HR-M cameras, and 16 synchronized low resolution grayscale frames $I_q^{hsm_j}$ ($q = 16i, \ldots, 16i+15$; $j = 0, 1$) captured by each of the two HS-M cameras. Since $I_p^{hrm_j}$ images are captured with large apertures to gather more light at the same period of time, they contain low level of noise. This indicates that they

will provide high quality gradient priors for denoising HS-M images. In this Chapter, we show how to utilize this information to enhance the high frequency details in the denoised results. In particular, we show that such high frequency details cannot be synthesized by state-of-the-art single image denoising methods such as BM3D [21].

Due to the difference in brightness and noise levels between HS-M and HR-M images, it would be difficult to directly locate corresponding patches as priors from HR-M images for any given patch in HS-M images. We tackle this problem by first preprocessing the HS-M images to improve the intensity level and then apply BM3D denoising algorithm to partially remove sensor noise, as shown in Fig.5.1. Specifically, we present a novel single image exposure fusion technique can boost the intensity level of the dark regions while avoiding the bright regions saturating. Compared with regular Gamma correction, this technique allows the BM3D denoised HS-M images to recover many important details as shown in Fig.5.2.

In the second stage, we locate low-noise patch priors from the preprocessed HR-M images for denoising the HS-M images. In Chap.4, we show that we can fuse an all-in-focus image from the HR-M image pair using our dual focus stereo imaging technique. Conceptually, we can then register the noisy HS-M image pairs with the fused result. In practice, accurately registering every patch is challenging due to strong image noise and fusion errors. Therefore, instead of applying direct warping, we propose a multi-view block matching scheme for finding patch correspondences between the HR-M and HS-M images.

Finally, we present an iterative optimization algorithm to denoise the preprocessed HS-M images. Based on the image noise model Eq.(5.1), we treat the image denoising problem as optimization, with the goal to minimize $\|I - \hat{I}\|^2$ with additional regularization terms. To take advantage of the heterogenous types of cameras in our hybrid camera system, we design a spatial prior from HR-M images to regularize $\|I - \hat{I}\|^2$, together with the conventional ℓ_1 total variation regularization term. Experiments show our multispectral denoising algorithm is able to add back high frequency details to HS-M images.

5.3 HS-M Image Preprocessing: Low Dynamic Range Boosting

As was discussed in Sec.3.2, HS-M cameras use fast shutters and small apertures to avoid object motion blurs and defocus blurs. However, the images are under-exposed and hence can be highly noisy. The simplest method to enhance the image is to apply Gamma correction to reveal details in dark regions without saturating well-exposed regions. However, Gamma correction requires choosing proper parameters for finding the ideal camera response curve, in which the parameters are usually scene-dependent. Moreover, Gamma correction can often amplify the noise level and produce non-natural looking images.

Our solution is to extend the exposure fusion technique [82] to synthesize virtual exposures from the HS-M images. Recall that [82] uses multiple exposures whereas we only use a single image. Therefore, we first synthesize a sequence of virtual exposure images $I_i, (i = 1, \ldots, m)$ from a single image I_0 by amplifying its intensity m times. Next we compute a set of weight maps $W_i, (i = 1, \ldots, m)$ corresponding to each synthesized virtually exposed image I_i for fusion. The weight at pixel x is computed as the multiplication of the saturation cost and the local contrast cost. We use the Euclidean distance between the pixel value $I(x)$ ($I(x) \in [0, 1]$) and an intermediate intensity value κ to model the degree of pixel saturation, and use the Laplace filtered result as the local contrast measure. Finally, we fuse an output image $\tilde{I}(x)$ using pyramid fusion [15] of I_i with corresponding weight map W_i. Detailed algorithm can be found in Alg.5.1.

We define the saturation cost at pixel x as $Ws_i(x) = \exp((I_i(x) - \kappa)^2/\iota$, where $\kappa = 0.4$ and $\iota = 0.2$. To generate satisfactory fusion result, we empirically set the multiplier α for synthesizing I_i as 0.3 and $m = 4$. In theory, an optimal value of α could be estimated by analyzing the histogram distribution of I_0 when m is given. Fig.5.2 shows the low dynamic range boosted HS-M image. Our single image virtual exposure fusion technique is able to greatly enhance the dark regions.

It is worthy mentioning that more sophisticated fusion techniques such as the ones used in high dynamic range (HDR) imaging [23, 98] can be used in place of our fusion scheme. However HDR imaging not only requires estimating the camera response curve but also requires using complex algorithms such as tone mapping or gradient fusion [27, 32, 76,

Algorithm 5.1: Virtual exposure fusion

input : an under-exposure image $I_0(x)$
output: image $\tilde{I}(x)$ with intensity boosted

synthesize m additional images I_i according to multiplier α

$$I_i(x) = I_0(x) \times (1 + i \times \alpha), (i = 1, \ldots, m)$$

for $j \leftarrow 0$ **to** m **do**

  compute weight map $W_i(x)$ with respect to $I_i(x)$

$$W_i(x) = \text{Laplace}(I_i(x)) \times Ws_i(x),\ \forall x \in \mathbf{X}$$

for each pixel location x, normalize weight maps by

$$W_i(x) = W_i(x)/\Sigma_{i=0}^{i=m} W_i(x)$$

return the image $\tilde{I}$ by fusing the Laplacian pyramids of I_i level by level according to the Gaussian pyramids of weight maps W_i, $(i = 0, \ldots, m)$

97]. In contrast, our virtual exposure fusion technique is based on pyramid fusion, therefore it is much easier to use and is much faster.

5.4 Multi-view Block Matching

Next, we present a multi-view block matching (MVBM) technique for patch-based image denoising. The goal of MVBM is to find similar patches from HR-M image pairs for each patch in HS-M images. This is a challenging problem for a number of reasons. Firstly, HR-M image pairs can still have strong defocus blurs in various regions. As was discussed in Sec.3.2, our hybrid camera system uses only two large aperture HR-M cameras. As a result, not all depth layers can be clearly focused. Secondly, the noise level of HS-M images is much more severe than that of HR-M images. Even after we apply the virtual exposure based preprocessing, direct block matching between HS-M and HR-M images is difficult.

Recall that in our hybrid camera system, the resolution of HS-M images $I_i^{hsm_j}$ $(j = 0, 1.)$ images are lower than that of HR-M images $I_{\lfloor i/4 \rfloor}^{hrm_j}$. Therefore, we down-sample the HR-M images to the same resolution as HS-M images. We use $I_{\lfloor i/4 \rfloor}^{hrm_j}$ to represent the down-sampled HR-M images where $\lfloor \cdot \rfloor$ denotes the floor operator.

We fix the size of each image block B_x to be $N_s \times N_s$ where its top left element is at position x. For any given block B_x in image $I_i^{hsm_j}$, we search both $I_{\lfloor i/4 \rfloor}^{hrm_0}$ and $I_{\lfloor i/4 \rfloor}^{hrm_1}$ to

Figure 5.2: The preprocessing of low-light images. (left) A set of images with different exposure time synthesized from a typical HS-M image captured under low-light conditions. (center) is our preprocessing result of the input HS-M image. Notice that the intensity level of dark regions has been boosted while the bright regions are still well under saturation. (right) shows the BM3D denoising result of the preprocessed HS-M image.

find its similar blocks. We can also consider the block similarity in the temporary domain besides the spatial domain to improve the accuracy of block matching. For example, we can search stereo pairs $(I^{hrm_0}_{\lfloor i/4 \rfloor -1},\ I^{hrm_1}_{\lfloor i/4 \rfloor -1})$, $(I^{hrm_0}_{\lfloor i/4 \rfloor},\ I^{hrm_1}_{\lfloor i/4 \rfloor})$, and $(I^{hrm_0}_{\lfloor i/4 \rfloor +1},\ I^{hrm_1}_{\lfloor i/4 \rfloor +1})$ to find similar patches of B_x. In our work, we only consider one stereo pair $(I^{hrm_0}_0, I^{hrm_1}_0)$.

We use the sum of squared distance (SSD) to measure the similarity between different image patches. In practice, directly computing the SSD between two image patches in spatial domain would introduce great errors due to the existence of image noise. Therefore, we first transform 2D image patches into frequency domain, then threshold the coefficients to reduce the noise level, and finally apply the SSD matching in the frequency domain. Specifically, we first apply the state-of-the-art block matching with 3D filtering (BM3D) [21] algorithm to reduce the noise level of the HS-M images. Next, we can directly apply the multi-focus fusion technique presented in Chap.4 to generate an all-in-focus image from the image pair $(I^{hrm_0}_{\lfloor i/4 \rfloor},\ I^{hrm_1}_{\lfloor i/4 \rfloor})$, which can be used to provide gradient priors for HS-M image denoising through SSD matching between them.

To further improve the performance, we propose an alternative approach. Since our

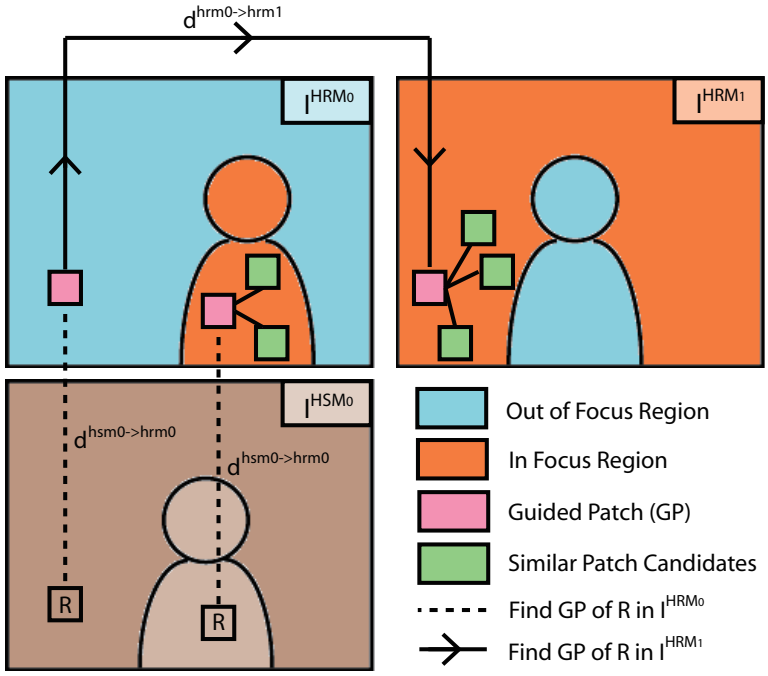


Figure 5.3: Multiview block matching. We use the defocus kernel map estimated from the HR-M pair to decide whether the patch candidate is in-focus or defocused. And we use the disparity map $d^{hsm_0 \to hrm_0}$ estimated from I^{hsm_0} and I^{hrm_0} to guide the multiview block matching.

multispectral denoising method is a patch-based optimization problem, we only need to extract in-focus gradient priors from HR-M pairs for any given patch in HS-M images. This indicates that we can bypass the all-in-focus image synthesization and directly use SSD matching to find the in-focus patches, as long as we can differentiate which parts are in-focus and which parts are defocus blurred. Therefore, our solution is to first apply the blur disk size equation, Eq.4.6, to find that specific disparity value $\gamma^\star$ which will lead to equal defocus blurs on both images. Next we apply the defocused stereo matching technique presented in Sec.4.2 to estimate a defocus kernel map from the HR-M pair. We use $\gamma^\star$ as the threshold to find in-focus patches from HR-M pairs based on the estimated defocus kernel map or disparity map. In practice, we approximate the equal blur disparity value $\gamma^\star$ from the average of the two in-focus disparity values with respect to the two HR-M images.

Fig.5.3 illustrates applying our MVBM algorithm for image I^{hsm_0}. Similar approach

applies to I^{hsm_1}. We first map the reference block B_x to image I^{hrm_0} through disparity map $d^{hsm_0 \to hrm_0}$. If the guided patch is within in-focus regions of image I^{hrm_0}, we just search locally to find similar candidate patches for B_x. If the guided patch is within the defocus regions, we map this guided patch to image I^{hrm_1} through disparity map $d^{hrm_0 \to hrm_1}$, and then search locally in image I^{hrm_1} for similar patches of B_x. In the last case when the guided patch is located on the boundary of in-focus regions, we use the majority of disparity values covered by the image patch to decide whether we should search locally or map it to image I^{hrm_1}.

5.5 Multispectral Denoising

In this section, we show how to use the patch correspondences between HR-M images I^{hrm_j} and HS-M images I^{hsm_j} for multispectral denoising of I^{hsm_j}. Since the image sensors used by HR-M cameras are capable to capture a wide range of wavelength including near infrared (NIR) and visible light, we call our denoising algorithm a multispectral method.

5.5.1 Problem Formulation

Based on the image noise formulation(Eq. (5.1)), we design an optimization algorithm to recover the latent image $\hat{I}$ from the noisy observation I. Since we assume the noise η is additive and Gaussian, the data fidelity term takes the ℓ_2-norm, defined as $\|I - \hat{I}\|_2^2$. As image denoising is one of the classical ill-posed inverse problems, we add a total variation (TV) regularization [100, 101] term to the data fidelity term to make the reconstruction process more reliable, e.g., to overcome the drawbacks of Tikhonov-like regularization [116] and preserve high frequency details such as sharp edges, patterns, and fine textures.

It is important to note that the total variation regularization alone does not help recover additional high frequency details of the original under-exposed image. Together with the data fidelity term, they can only partially remove sensor noise from images. Therefore, we design a new multispectral spatial regularization that uses patches from the HR-M cameras $\hat{I}^{hrm}$ to add back the high frequency details.

In all, we formulate the multispectral denoising for HS-M images as:

$$\min_{\hat{I}^{hsm_j}} \underbrace{\frac{1}{2}\|\hat{I}^{hsm_j} - I^{hsm_j}\|_2^2}_{Data\ Fidelity} + \underbrace{\mu_1\|D\hat{I}^{hsm_j}\|_1}_{Total\ Variation} + \underbrace{\mu_2\|D\hat{I}^{hsm_j} - D\hat{I}^{hrm^\star}\|_1}_{Spatial\ Prior}, \tag{5.2}$$

where $j = \{0, 1\}$; μ_1 and μ_2 are two positive weighting terms used to control the strength of TV and serve as spatial prior regularization respectively; $\|\cdot\|_1$ uses the ℓ_1-norm; $D = (D^{(1)}; D^{(2)}) \triangleq (D^{(1)^\top}, D^{(2)^\top})^\top$ where $D^{(1)}$ and $D^{(2)}$ stand for the first-order finite difference operator in horizontal and vertical directions respectively and $^\top$ is the transpose operator. $D^{(1)}$ and $D^{(2)}$ are the $MN \times MN$ convolution matrices formed from two 1D derivative filters $d_1 = [1, -1]$ and $d_2 = [1, -1]^\top$. Image $\hat{I}^{hrm^\star}$ is obtained by first downsampling the fused all-in-focus image $\hat{I}^{hrm}$, and then registering with image I^{hsm_j}. All images are treated as column vectors. For the sake of clarity, we remove the subscripts for each image in Eq.(5.2).

One key challenge in solving Eq. (5.2) is patch matching between $\hat{I}^{hrm^\star}$ and I^{hsm_j}. Due to image noise, defocus blurs, and the noise level differences, it is very difficult to do image registration between them, even with the recent advances of optical flow methods [107, 108, 132] and stereo matching methods [57, 124]. Therefore in this section, we design a new patch-based optimization algorithm to denoise the HS-M image pairs. To utilize the patch priors from HR-M images, we modify Eq.(5.2) as

$$\min_{\hat{B}_x} \frac{1}{2}\|\hat{B}_x - B_x\|_2^2 + \mu_1\|D\hat{B}_x\|_1 + \mu_2\|D\hat{B}_x - DB_x^{hrm^\star}\|_1, \forall x \in \mathbf{X}, \tag{5.3}$$

where $B^{hrm^\star}$ is obtained by using our MVBM method. For simplicity, in Eq.(5.3) we use $(B_x,\ \hat{B}_x)$ to represent $(B_x^{hsm_j},\ \hat{B}_x^{hsm_j},\ j = \{0, 1\})$, i.e., image patches extracted from Image $(I^{hsm_j},\ \hat{I}^{hsm_j})$. The final estimation of $\hat{I}^{hsm_j}$ can be obtained from patch estimates $\hat{B}_x$ $(x \in \mathbf{X})$ as

$$\hat{I}^{hsm_j} = \frac{\sum_{x\in\mathbf{X}} w_x \hat{Y}_x}{\sum_{x\in\mathbf{X}} w_x \mathcal{X}_x}, \tag{5.4}$$

where $\hat{Y}_x$ has the same size as image $\hat{I}^{hsm_j}$ and is zero-padded outside of block estimate $\hat{B}_x$, $\mathcal{X}_x : \mathbf{X} \leftarrow \{0, 1\}$ is the characteristic function of the square support of a block $\hat{B}_x$ with its

top left element located at $x \in \mathbf{X}$, and w_x is the weight window for aggregating all the image patches. Similar to [21], we use a $N_s \times N_s$ Kaiser-Bessel window as the patch aggregation weight to reduce border effects.

5.5.2 Iterative Optimization

Compared with Tikhonov-like regularization, the minimization problem formulated by Eq.(5.3) is computationally expensive to solve due to non-linearity and non-differentiability of the TV regularization and the spatial prior. Inspired by the half-quadratic penalty method [36,37] and alternating optimization method [114,123], we present an effective solution: we introduce two auxiliary vectors $u, v \in \mathbb{R}^{2N_s^2}$ to the objective function in Eq. (5.3) and then reformulate this denoising problem equivalently as a constrained optimization problem with linear constraints,

$$\begin{aligned} \min_{\hat{B}_x,u,v} \ & \frac{1}{2}\|\hat{B}_x - B_x\|_2^2 + \mu_1\|u\|_1 + \mu_2\|v\|_1 \ , \forall x \in \mathbf{X}, \\ \text{s.t.} \ & \begin{cases} u - D\hat{B}_x = 0, \\ v - (D\hat{B}_x - DB_x^{hrm^\star}) = 0. \end{cases} \end{aligned} \tag{5.5}$$

Notice that we now eliminate the non-differentiable terms, i.e., the ℓ_1 TV regularization and spatial prior and make the objective function in Eq.(5.5) separable and the constraints linear. When either two of the three variables $\hat{B}_x$, u and v are fixed, minimizing the objective function with respect to the other has a closed-form formula with low computational complexity and strong numerical stability.

To solve for the Eq.(5.5), we apply an alternating minimization method [114, 123] to the augmented Lagrangian function [40,46] of Eq.(5.5) using the following iterative scheme:

$$\hat{B}_x^{k+1} \leftarrow \arg\min_{\hat{B}_x} \mathcal{L}_{\mathcal{A}}(\hat{B}_x, u^k, v^k, \lambda_1^k, \lambda_2^k), \tag{5.6a}$$

$$u^{k+1} \leftarrow \arg\min_{u} \mathcal{L}_{\mathcal{A}}(\hat{B}_x^{k+1}, u, v^k, \lambda_1^k, \lambda_2^k), \tag{5.6b}$$

$$v^{k+1} \leftarrow \arg\min_{v} \mathcal{L}_{\mathcal{A}}(\hat{B}_x^{k+1}, u^{k+1}, v, \lambda_1^k, \lambda_2^k), \tag{5.6c}$$

$$\lambda_1^{k+1} \leftarrow \lambda_1^k - \frac{\gamma}{\beta_1}(u^{k+1} - D\hat{B}_x^{k+1}), \tag{5.6d}$$

$$\lambda_2^{k+1} \leftarrow \lambda_2^k - \frac{\gamma}{\beta_2}\Big(v^{k+1} - (D\hat{B}_x^{k+1} - DB_x^{hrm^\star})\Big), \tag{5.6e}$$

where $\mathcal{L}_{\mathcal{A}}(\hat{B}_x, u, v, \lambda_1, \lambda_2)$ is the augmented Lagrangian function of Eq.(5.5) defined by

$$\begin{aligned}\mathcal{L}_{\mathcal{A}}(\hat{B}_x, u, v, \lambda_1, \lambda_2) \triangleq\ & \frac{1}{2}\|\hat{B}_x - B_x\|_2^2 + \mu_1\|u\|_1 + \mu_2\|v\|_1 \\ & - \lambda_1^\top(u - D\hat{B}_x) - \lambda_2^\top(v - (D\hat{B}_x - DB_x^{hrm^\star})) \\ & + \frac{1}{2\beta_1}\|u - D\hat{B}_x\|_2^2 + \frac{1}{2\beta_2}\|v - (D\hat{B}_x - DB_x^{hrm^\star})\|_2^2\,,\end{aligned} \tag{5.7}$$

where λ_1 and λ_2 are two vectors of Lagrange multipliers and $\lambda_1, \lambda_2 \in \mathbb{R}^{2N_s^2}$, β_1 and β_2 are penalty parameters and γ is the step length for updating λ_1 and λ_2. By iteratively updating the Lagrange multipliers λ_1 and λ_2, the solution to Eq.(5.6a)-(5.6c) will eventually converge to the one to Eq.(5.5). Notice that our minimization problem can also be solved by other ℓ_1 solvers, for example, the fast iterative shrinkage-thresholding algorithm (FISTA) [6].

Eq.(5.6a)-(5.6c) also shows that the original optimization problem Eq.(5.3) is now separable with respect to $\hat{B}_x$, u, and v and thus can be divided into 3 sub-problems. In our case, we design an iterative algorithm to reconstruct the noise-free image $\hat{I}^{hsm_j}$ by alternately minimizing $\hat{B}_x$, u and v. λ_1 and λ_2 are directly updated by Eq.(5.6a)-(5.6e). Details of our patch-based multispectral image denoising algorithm can be found in Alg. 5.2.

5.5.2.1 $\hat{B}_x$ Sub-problem

In our framework, when variables u, v, λ_1 and λ_2 are fixed and have values from the previous iteration k, we can simplify Eq.(5.5) to a quadratic problem in $\hat{B}_x$. Using Lagrange

Algorithm 5.2: Patched based multispectral image denoising

input : Noise patch B_x from HS-M images, and its registered spatial prior $B_x^{hrm^\star}$

output: Denoised image patch $\hat{B}_x$

$\hat{B}_x^0 = B_x$, $k = 0$

while $k < iter_{max}$ && $\frac{\|\hat{B}_x^{k+1} - \hat{B}_x^k\|}{\|\hat{B}_x^k\|} > \epsilon$ **do**

  given fixed u^k, v^k, λ_1^k, and λ_2^k, solve Eq.(5.9) to give $\hat{B}_x^{k+1}$

  given fixed $\hat{B}_x^{k+1}$, v^k, λ_1^k and λ_2^k, solve Eq.(5.11) to give u^{k+1}

  given fixed $\hat{B}_x^{k+1}$, u^{k+1}, λ_1^k and λ_2^k, solve Eq.(5.12) to give v^{k+1}

  solve λ_1^{k+1} by Eq.(5.6d)

  solve λ_2^{k+1} by Eq.(5.6e)

  $k = k + 1$

return denoised image patch $\hat{B}_x$

multipliers, we have the necessary condition to find the optimal $\hat{B}_x$ for iteration $k+1$,

$$\nabla_{\hat{I}} \mathcal{L}_{\mathcal{A}}(\hat{B}_x, u^k, v^k, \lambda_1^k, \lambda_2^k) = 0, \tag{5.8}$$

which gives us the optimal $\hat{B}_x^{k+1}$:

$$\Big[(\frac{1}{\beta_1} + \frac{1}{\beta_2})D^\top D + E\Big]\hat{B}_x^{k+1} = B_x + D^\top(\frac{u^k}{\beta_1} + \frac{v^k + DB_x^{hrm^\star}}{\beta_2} - \lambda_1^k - \lambda_2^k)\,, \tag{5.9}$$

where E is the identity matrix of size $N_s^2 \times N_s^2$. Solving the large linear system indicated by Eq.(5.9) could give us the optimal $\hat{B}_x$ for iteration $k+1$. Notice that under the periodic boundary conditions for $\hat{B}_x$, the solution to Eq.(5.9) can also be computed efficiently in the 2D frequency domain by FFT division [123].

5.5.2.2 u Sub-problem

From Eq.(5.6a), when given fixed variables $\hat{B}_x^{k+1}$, v^k, λ_1^k and λ_2^k, we can update u by minimizing the following optimization problem

$$\min_u \frac{1}{2\beta_1}\|u - (D\hat{B}_x^{k+1} + \beta_1\lambda_1^k)\|_2^2 + \mu_1\|u\|_1\,. \tag{5.10}$$

Notice that the regularization term of Eq.(5.10) is in ℓ_1-norm, we can have its unique minimizer through two dimensional shrinkage as

$$u^{k+1} = \mathrm{sgn}(D\hat{B}_x^{k+1} + \beta_1\lambda_1^k) \circ \max\left\{|D\hat{B}_x^{k+1} + \beta_1\lambda_1^k| - \mu_1\beta_1, 0\right\}, \tag{5.11}$$

where operator sgn denotes the signum function, and operator $\circ$ represents the pointwise product.

5.5.2.3 v Sub-problem

Finally, when given fixed variables $\hat{B}_x^{k+1}$, u^{k+1}, λ_1^k and λ_2^k, similar to the u sub-problem, we have the solution of optimal v for iteration $k+1$ as

$$\begin{aligned} v^{k+1} =& \mathrm{sgn}(D\hat{B}_x^{k+1} - DB^{hrm^\star} + \beta_2\lambda_2^k) \\ & \circ \max\left\{|D\hat{B}_x^{k+1} - DB^{hrm^\star} + \beta_2\lambda_2^k| - \mu_2\beta_2, 0\right\}. \end{aligned} \tag{5.12}$$

In practice, we assign a parameter γ to control the step size of the updating equations for λ_1 and λ_2. For better convergence of our algorithm, we choose $\gamma \in (0, 2)$. The termination criteria of Alg.5.2 requires either the maximum number of iterations in the optimization process reached or the relative difference between the successively reconstructed image values well below some predefined tolerance $\epsilon > 0$. In our experiments, we let $\beta_1 = \beta_2$.

5.6 Results and Discussion

We evaluate our algorithms on both synthetic and real images to demonstrate their robustness and effectiveness.

Synthetic Scenes. To quantitatively compare our hybrid denoising algorithm with BM3D, we render a set of images using Autodesk© 3ds Max©. We create an animation for the front paper cup, focus one HR-M camera on the front table, and the other one on the background orchid, as shown in Fig.5.4. We set the framerate and resolution of each camera according to our hybrid camera setup (Chapter 3), except for the HR-C camera. To simulate the motion blur, we render 16 frames from the HR-C camera and compute the average of them as the motion blurred HR-C image. To synthesize noisy HS-M and HR-C images, we add to them with zero-mean Gaussian noise of variance 0.09 and 0.03 respectively.

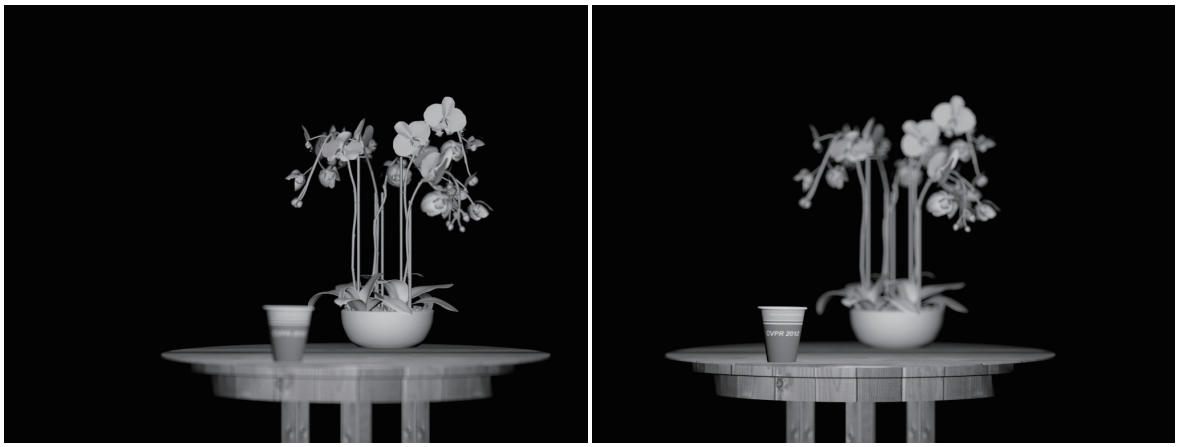


Figure 5.4: One sample of synthetic HR-M image pair.

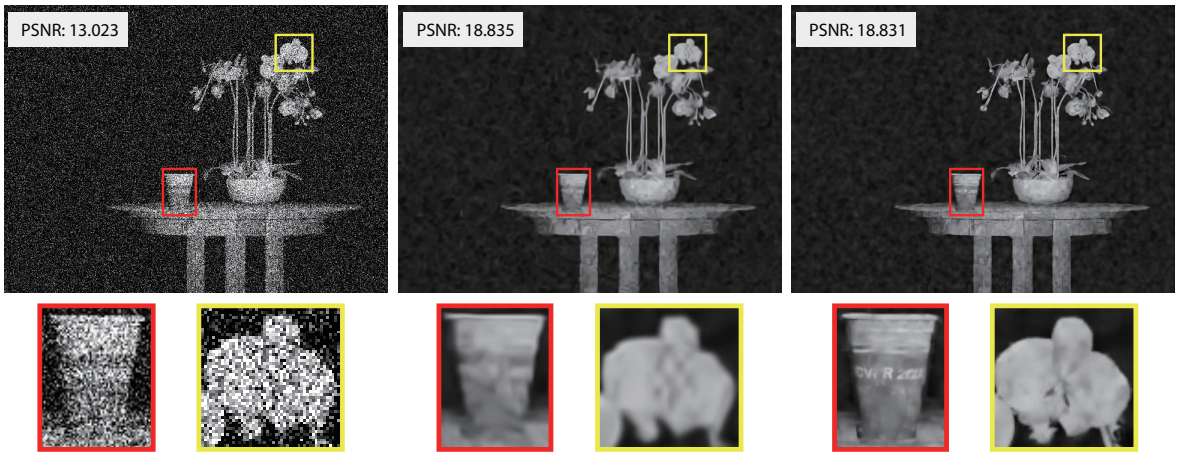


Figure 5.5: Multispectral denoising of a synthetic scene. (left) the synthesized low-light image captured by one HS-M camera. (center) the BM3D denoising result. (right) our result.

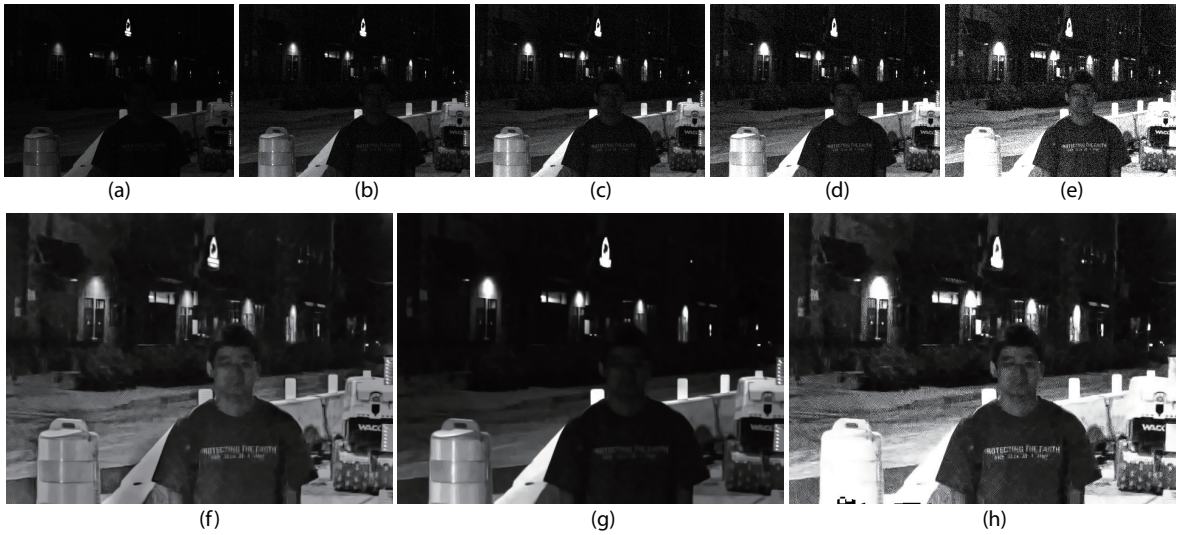


Figure 5.6: Another example of our exposure fusion technique. (a) A typical image from HS-M cameras captured under low-light conditions. We then synthesize another four additional images (b-e) according to multiplier α. We apply Alg.5.1 to fuse an image from (a-e) with the intensity of dark regions boosted and bright regions still under saturated. (f-h) show the BM3D denoising results of that exposure enhanced image, (c), and (e) respectively. (The subject depicted here is the author of this thesis.)

In Fig.5.5, we show our multispectral denoising result of a synthesized HS-M image, and compare it with BM3D method in terms of peak signal-to-noise ratio (PSNR) in decibels (dB). Even though BM3D outperforms our algorithm by 0.004 dB, our result acctually looks better and contains much more details, such as the texture of the table, the characters on the paper cup, and the shape of background orchid. To achieve best denoising quality, we set $\mu_1 = 0.1$ and $\mu_2 = 0.1$ for both synthetic and real scenes.

Real Scenes. In Fig.5.6, we show another example for our virtual image exposure fusion technique. Notice that the input low-light image (a) is very challenging for denoising since it exhibits extremely low contrast and is almost black except at some saturated road lights. Therefore we first synthesize a set of images (b-e) from (a) with different boosting levels, and then apply our exposure fusion method discussed in Sec.5.3 to synthesis a well exposed image for BM3D denoising. As can be seen from the figure, the BM3D denoised result (f) of that exposure enhanced image can display many details of the scene, such as the

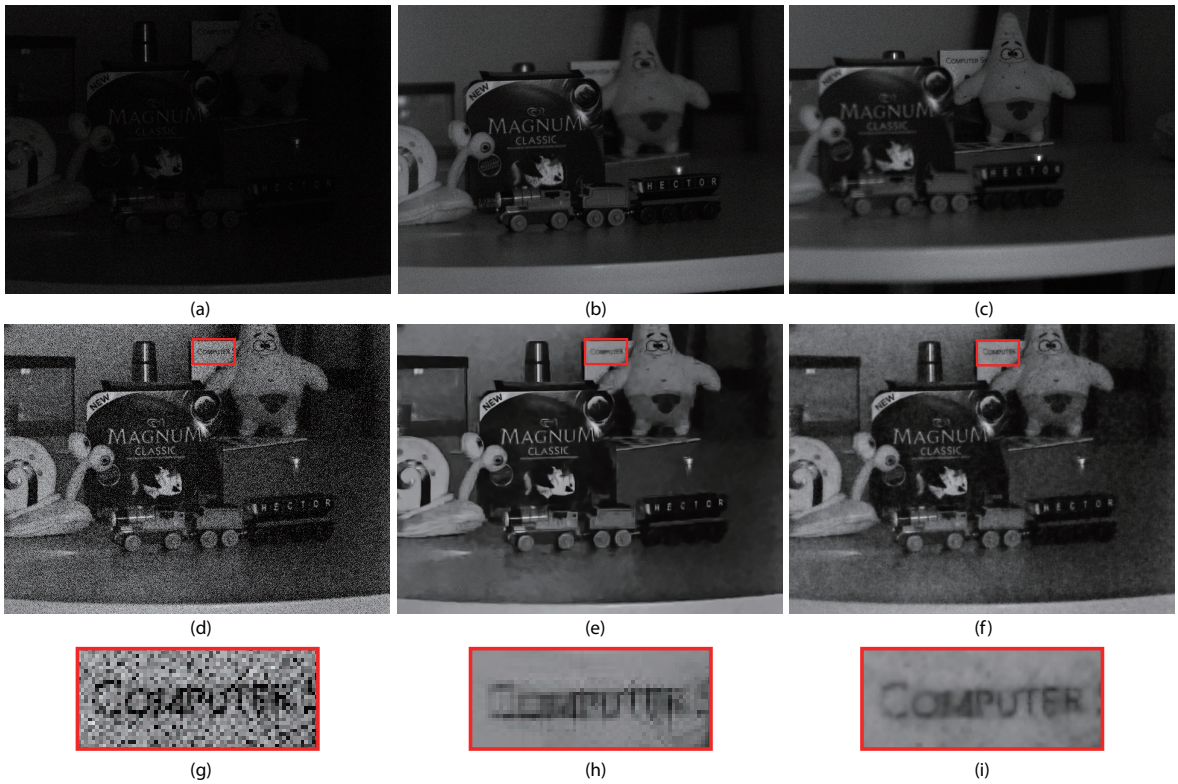


Figure 5.7: Multispectral denoising of an indoor scene. (a) is a low-light image captured by camera HS-M_0, and (b)(c) are large aperture stereo images from HR-M cameras. We first synthesis a well exposed image (d) from (a) for image denoising. (e)(f) shows the denoising results of (d) by BM3D method and our multispectral denoising algorithm respectively. (g-i) are zoomed-in regions of the corresponding rectangle regions shown in (d-f).

brand name of the construction machine and the background restaurant. It also has much better overall quality than (g) and (h). For instance, the road work region (right side of the image) and the foreground people are well exposed in (f), while (g) and (h) suffer from strong artifacts of either under exposure or over saturation.

Fig.5.7 shows an indoor scene of a toy train moving in front of a cluttered background. We set the focus of one HR-M camera at the front toy train and chocolate box, while the other one focuses at the background book, which can be clearly identified from the strong defocus blurs in (b) and (c). By equipping HR-M cameras with large aperture lenses, these

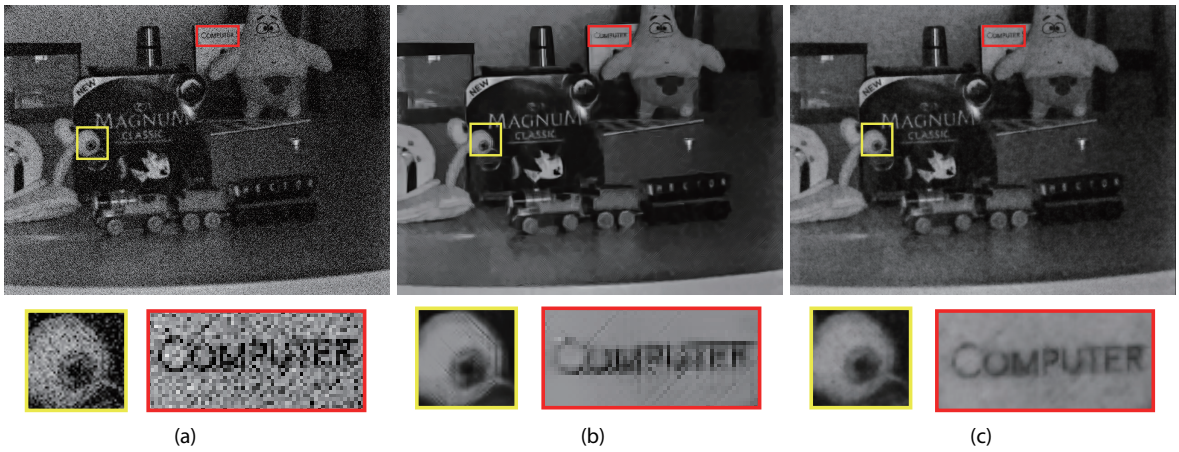


Figure 5.8: Multispectral denoising of the same indoor scene as Fig.5.7 for the other HS-M camera. (left) the preprocessed low-light image from the second HS-M camera. (center) the BM3D denoising result of (left). (right) our multispectral denoising result of (left). The second row shows zoomed-in regions of the corresponding rectangle areas in the first row.

HR-M images contain nearly no noise, which makes them suitable for regularizing the objective function of multispectral denoising. We first synthesis a low dynamic range image (d) out of the original HS-M image (a) to bring in global details of the scene, but at the expense of introducing strong sensor noise. As can be seen in the zoomed-in region (g), the image (d) is greatly degraded due to the bad illumination of the scene. With the regularization of the ℓ_1 TV term and our multispectral prior, our algorithm can fully recover these local details, such as the title of the background book, and the contour of the foreground train, while BM3D algorithm fails in these regions. In addition, the denoising result of BM3D algorithm looks too smooth and has artifacts along boundaries or edges.

To demonstrate the quality of our denoised results, we run some on-line Optical Character Recognition (OCR) softwares (for example, http://www.onlineocr.net) on Fig.5.7(i). The OCR algorithm can accurately recognize all the letters in Fig.5.7(i)[1], while fails to recognize any character in Fig.5.7(h) even after sophisticated image adjustment. Fig.5.8 demonstrates our denoising result of the same scene as Fig.5.7 for the other HS-M camera. Notice

[1] The only thing we need to do is to increase the image contrast by simply adjusting curves.

Figure 5.9: Multispectral denoising of an outdoor scene. (a) is a low-light image captured by camera HS-M_0, and (b)(c) are large aperture stereo images from HR-M cameras. (d) is the preprocessed image from (a), and (e)(f) are the denoising results of (d) by BM3D and our multispectral denoising algorithm respectively. (g-i) are zoomed-in regions of the corresponding rectangle regions shown in (d-f). (The subject depicted here is the author of this thesis.)

the strong parallax between these two views (5.7(f),5.8(c)). In Fig.5.8(b), we can see that BM3D method is greatly affected by the diagonal scan line noise pattern[2] and has strong artifacts in the zoomed-in regions, while our method can correctly recover both the title of the background book and the eye of the snail.

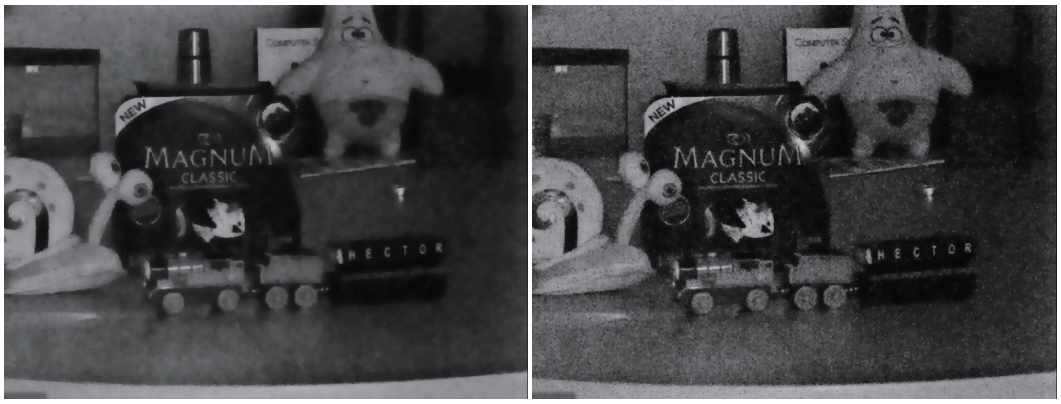
Compared to indoor scenarios, it is much more difficult to denoise outdoor low-light images. They usually contain much wider dynamic range than day light images. When using current standard digital imaging techniques or photographic methods, it is inevitable to create images of poor quality. For instance, the image shown in Fig.5.9 (a) has multiple

[2] This noise pattern is possibly caused by rolling shutter.

Figure 5.10: Multispectral denoising of the same outdoor scene as Fig.5.9 for the other HS-M camera. (left) the preprocessed low-light image from the second HS-M camera. (center) the BM3D denoising result. (right) our multispectral denoising result. The second row shows zoomed-in regions of the corresponding rectangle areas in the first row. (The subject depicted here is the author of this thesis.)

overexposed regions, e.g., the road lights and the headlights of the background SUV, while the rest are dark regions and exhibit extremely low contrast. From fig.5.9 (e)(f), we can see that our multispectral denoising method can compete with the state-of-the-art denoising method BM3D at most regions. In some regions, such as zoomed-in regions shown in (h), our method performs even better than BM3D. It can recover fine details of the scene, e.g., characters on the person's T-shirt, which could be potentially useful for many other night time applications. Fig.5.10 demonstrates our denoising result of the same scene for the other HS-M camera. When taking a close look at the zoomed-in region of Fig.5.10(c), our method can maintain the structure of the black tire, which is destroyed in BM3D's result (b).

We also want to point out that patch based total variation (TV) method usually has better denoising results than plain TV method in terms of visual appearance. If we set $\mu_2 = 0$ in both Eq.(5.2) and Eq.(5.3), we can compare these two denoising methods without the multispectral regularization. As can be seen from Fig.5.11, patch based TV method can recover more local details, while TV denoising has much smoother global appearance. With the



(a) (b)

Figure 5.11: Comparision between total variation (TV) denoising and patch based TV denoising. (left) TV denoising result of Fig.5.7(d). (right) patch based TV denoising result.

multispectral regularization, our patch based denoising method can further improve the denoising results and bring in detailed information from HR-M images. Another key advantage of our patch based denoising method is that it can handle occlusion boundaries automatically, because of the multi-view structure of our hybrid camera. Under extreme low-light conditions, however, our multi-view block matching may fail to find corresponding patches for some HS-M image patches due to the poor quality of estimated disparity maps. In these cases, we choose to use the brute force way to find similar patches from the HR-M image pair and use the salient metric presented in Sec.4.2.1 to select in-focus patches as multispectral priors.

Running Time. We evaluate the computational performance of our multispectral denoising on a Windows© 7 64bit system with Intel© Core(TM) i7-2600K @3.40GHz, and compare it with our C implementation of the state-of-the-art denoising method BM3D [21]. It takes approximately 119.122 seconds for our algorithm to denoise a HS-M image of size 640×480 when corresponding multispectral priors are available, while BM3D algorithm takes only 2.442 seconds to denoise an image of the same size. For multi-view block matching, most of the computation time is spent on the estimation of disparity maps, because of the strong parallax in both x and y directions of the camera grid. In some extreme low-light

cases, the disparity maps estimated are not accurate enough to guide the search of similar blocks between different views. Therefore, we choose to run block matching directly and it takes roughly 4.992 seconds on average to find all the necessary multispectral priors for denoising a HS-M image. Note that for best denoising quality, we use patch size 8×8, patch sliding step size 4 for BM3D method, and patch size 12×12, sliding step size 3 for our denoising algorithm. In our experiments, we use the OpenMP© API to improve the performance of our algorithm on multi-core systems.

5.7 Conclusion

In this chapter, we have presented a framework to remove sensor noise in low-light images captured by our hybrid camera system. We first preprocess the captured images from HS-M cameras using our exposure fusion technique. Then we design an novel optimization scheme that uses image patches from the low-noise HR-M images as the gradient prior for regularizing the data fidelity term of the objective function, together with the ℓ_1 TV term. Experiment results of both indoor and outdoor scenes show that our multispectral denoising algorithm is robust for extreme low-light conditions and can yield better results than the state-of-the-art single image denoising algorithm BM3D [21] for some important texture regions.

Chapter 6

MULTI-SPEED DEBLURRING

In this chapter, we demonstrate how to estimate the blur kernels for HR-C image deblurring using our hybrid camera. Different from the hybrid camera we proposed in Sec.3.2, we use a simplified hybrid camera here for easy demonstration of the multi-speed deblurring algorithm. It only consists a pair of high-speed color (HS-C) cameras and a single high-resolution color (HR-C) camera. The images are captured under normal light conditions, thus don't have strong sensor noises. We develop efficient algorithms to simultaneously motion-deblur the HR-C image and reconstruct a high resolution depth map. A high resolution depth map is extremly important for our hybrid camera system since it is the base to explore the geometric relationship between these cameras for our multispectral denoising and many other computer vision applications. Our method first estimates the motion flow in the HS-C pair and then warps the flow field to the HR-C camera to estimate the point spread function (PSF). We then deblur the HR-C image and use the resulting image to enhance the low-resolution depth map using joint bilateral filters. An example of the results generated by our hybrid camera is shown in Fig.6.1. Experiments show that our framework is robust and highly effective.

6.1 System Setup and Algorithm Overview

Our simplified hybrid camera uses a pair of PointGrey Dragonfly Express cameras as the HS-C cameras. The Dragonfly Express captures images of resolution 320x240 at 120 fps. We also position a PointGrey Flea2 camera on top of the two Dragonfly cameras. The Flea2 serves as the HR-C camera and captures images of resolution 1024x768 at 7.5 fps. We use the software solution provided by PointGrey to synchronize the HS-C cameras and the HR-C camera so that every HR-C frame synchronizes with 16 HS-C frames. These three

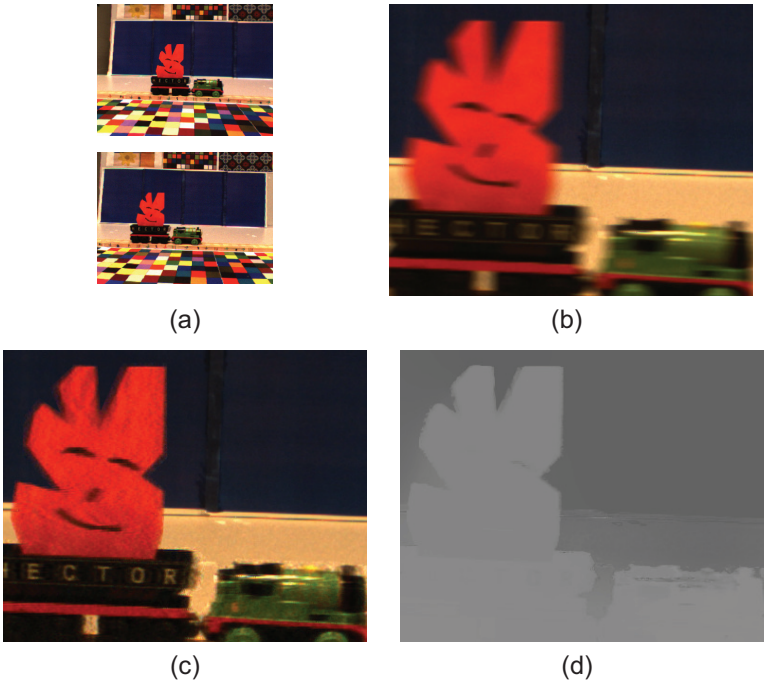


Figure 6.1: Motion deblurring and depth map super-resolution results using our hybrid camera. (a) shows two frames from the HS-C cameras, (b) shows a cropped region of the HR-C image. (c) shows the motion deblurred result. (d) shows the reconstructed high resolution depth map.

cameras are connected to two SIIG FireWire 800 3-Port PCIe cards. We attach all three camera modules to a wood plate and mount it on 2 tripods, as shown in Figure 6.2. Each camera uses a micro-lens with 4.8mm focal length. The two HS-C are positioned about 7.5cm away from each other and the HR-C camera is placed above the HS-C cameras.

An overview of our motion deblurring and depth map super-resolution framework is shown in Figure 6.3. We assume each frame I_0 in the HR-C camera maps to two HS-C sequence $S_1(t)$ and $S_2(t)$ ($t = 1...K$ and $K = 16$ in our case). We first estimate the motion flow fields M_1 and M_2 in S_1 and S_2 and warp them onto I_0 as M_0. Next, we estimate the PSF in I_0 from M_0 and apply the R-L algorithm to deblur I_0. Recall that I_0 corresponds to K consecutive HS-C frames, therefore, I_0 can be deblurred using K different PSFs, each derived from the relative motions with respect to a specific HS-C frame. We use $\tilde{I}_0(t)$ to represent the deblurred result of I_0 for frame t. To generate the super-resolution depth map,

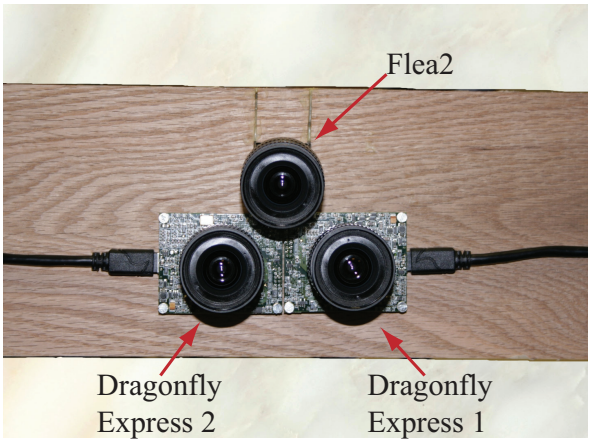


Figure 6.2: The prototype of our proposed hybrid camera. It consists of a rig of three cameras: a PointGrey Flea2 that serves as the HR-C camera and two PointGrey Dragonfly Express that serve as the HS-C cameras.

we first compute a low resolution depth map $D_L(t)$ between $S_1(t)$ and $S_2(t)$. We then warp $D_L(t)$ to the downsampled HR-C camera as $D_0^{'}(t)$ and then upsample $D_0^{'}(t)$ using a joint bilateral filter, whose spatial kernel is defined on $D_0^{'}(t)$ and range kernel defined on $\tilde{I}_0(t)$.

6.2 Motion Deblurring

In this section, we show how to use our hybrid camera for efficient motion deblurring.

6.2.1 Estimating Motion Flow

We first partition each frame in the HS-C camera into multiple foreground regions Ω_i^f and a background region Ω^b. To do so, we simply take a background image without any foreground objects and then use fore/background subtraction followed by graph-cut [13] to extract the foreground regions. We then group all foreground pixels into separate connected regions. We assume that each region has homogeneous motion and the regions do not overlap. We also use the estimated disparity map (Sec.6.3.1) to establish correspondences between the foreground regions in two HS-C cameras. This allows us to individually motion-deblur each foreground region using the method described in Sec.6.2.3. Notice that since our system captures a sequence of images, it is also possible to directly composite the background image by applying a median filter across the frames.

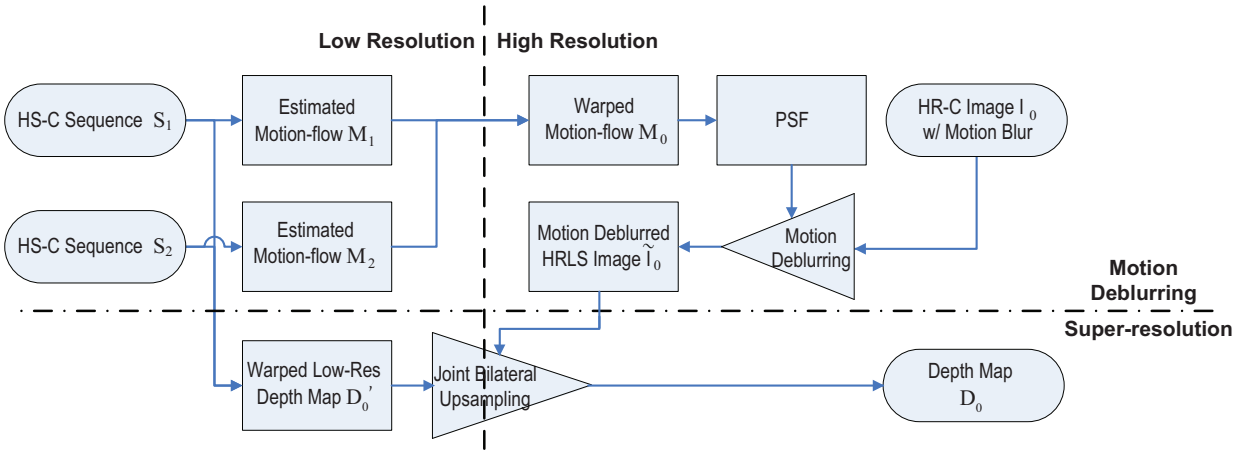


Figure 6.3: The processing pipeline using our hybrid camera for motion deblurring and depth map super-resolution. We first estimate the motion flows M_1 and M_2 in two HS-C sequences. We then warp the flow to the HR-C camera as M_0. To motion deblur the HR-C image, we estimate the PSF from M_0 and use the Richardson-Lucy algorithm to deblur the image. To construct a high resolution depth map D_0, we warp the low resolution depth map estimated from the HS-C cameras to the downsampled HR-C and use joint bilateral filters to upsample the D_0'.

To estimate the motion flow in each region Ω_i^f, we assume an affine motion model between two consecutive frames although we can concatenate several successive frames to form more complex motions. We apply a multi-resolution iterative algorithm [11] to minimize the energy function:

$$\arg\min_{p} \sum_{x} [I(w(x;p)) - I^{'}(x)]^2 \tag{6.1}$$

$$w(x;p) = \begin{pmatrix} p_1 x + p_2 y + p_3 \\ p_4 x + p_5 y + p_6 \end{pmatrix}$$

where p corresponds to the motion flow parameter, w is the warping function. In our case, we estimate the motion flow in each HS-C camera, i.e., $I = S_j(t)$ and $I' = S_j(t+1)$, $j = 1, 2$. At each frame, we use the Gauss-Newton method to find the optimal motion [4].

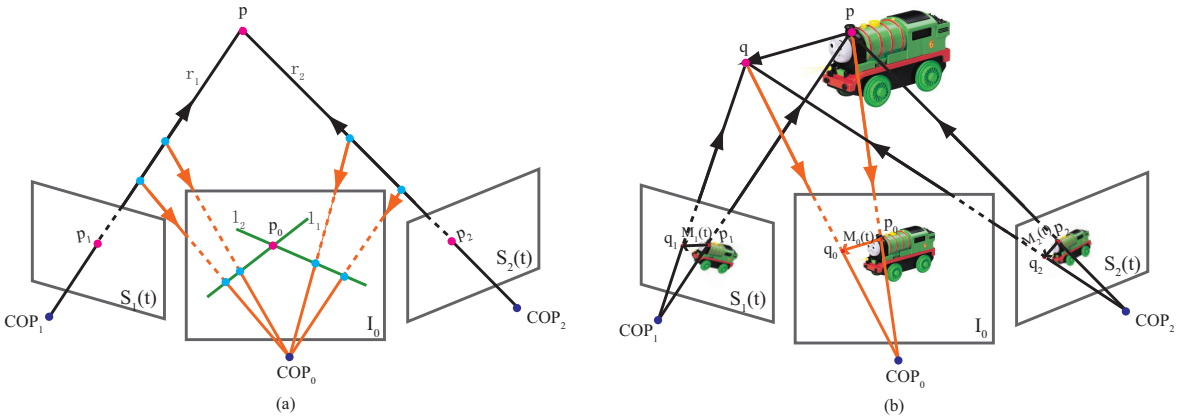


Figure 6.4: Motion estimation. (a) For each correspondence pair p_1 and p_2 in the HS-C pair, we project the ray that passes through p_1 in $S_1(t)$ onto I_0 as l_1. We similarly project p_2 as l_2 in I_0. Finally, we compute the intersection point of l_1 and l_2 to find the corresponding point p_0. (b) To estimate the motion flow of p_0 in I_0, we use the motion flow in the HS-C cameras to find q_0 and treat p_0q_0 as its motion flow.

6.2.2 Motion Warping

Once we estimate the motion flow for every foreground region Ω_i^f in each HS-C camera, we warp the the motion flow to the HR-C camera. Denote $M_1(t)$ and $M_2(t)$ as the estimated motion flow sample in S_1 and S_2 at frame t. To compute the motion flow sample $M_0(t)$ in the HR-C image I_0, we first establish correspondences between $S_1(t)$ and $S_2(t)$. We use SIFT feature detection [78] to process $S_1(t)$ and $S_2(t)$ and then perform a global matching. To remove outliers, our algorithm uses RANSAC with projective transformations as its precondition.

Given a pair of corresponding feature points p_1 and p_2 in $S_1(t)$ and $S_2(t)$, we connect p_1 with S_1 camera's center-of-project (COP) to form a ray r_1. We then project r_1 onto the HR-C image I_0 as a line l_1. We apply a similar process to p_2 to obtain line l_2. Finally we intersect l_1 and l_2 to obtain the p_0, as is shown in Figure 6.4(a).

To estimate the motion flow of p_0 in the HR-C camera I_0, we assume p_0's corresponding point p_1 in $S_1(t)$ moves to q_1 by motion flow $M_1(t)$ and p_2 in $S_2(t)$ moves to q_2 by $M_2(t)$. Similar to the way we find p_0, we then combine q_1 and q_2 to find their corresponding point

q_0 in I_0. We use the displacement between p_0 and q_0 as a motion flow sample. To improve robustness, we average multiple motion flow samples to compute $M_0(t)$.

Since our method combines the correspondences and the HS-C's motion flow, our camera is able to model more complex motions than the system proposed by Ben-Ezra and Nayar. In [7], the motion flow in the HR-C camera is assumed to be identical to the one estimated from the HS-C camera. This is only valid for planar motions. Our method separately estimates the motion flow for each foreground region Ω_i^f and warps it using the correspondences. Therefore, we can compute the motion flow for each foreground objects in I_0, even if they lie on different depth or move along different trajectories. Notice that the accuracy of our motion warping relies heavily on camera calibrations. In our implementations, we use the method in [145] to calibrate all 3 cameras.

6.2.3 PSF Estimation and Image Deconvolution

Recall that every HR-C frame maps to K $(K = 16)$ HS-C frames, therefore, we can compute the PSF with respect to each frame t. To do so, we concatenate all K discrete motion samples $M_0(t)$, $t = 1...K$ to form a single motion path. Ben-Ezra and Nayar proposed to use the motion centroid to construct a Voronoi tessellation for smoothly interpolating the motion path. Notice that our HS-C camera captures at a very high frame rate, therefore, each motion sample usually only covers 3 to 5 pixels in the HR-C image. To accurately estimate the kernel, we first appropriately align the motion path with the center of the kernel, then resample each $M_0(t)$ into N subpixels ($N = 50$ in our experiment), we further count, for each pixel p covered by $M_0(t)$ in the kernel, the number of subpixels N_p falling into p. Finally, we use this count N_p to estimate the weight of entry p in the kernel and normalize the kernel as the PSF.

Once we estimate the PSF, we can deblur the HR-C image using existing image deconvolution algorithms [34,138]. In our implementation, we choose to use the Richardson-Lucy (R-L) iterative deconvolution algorithm. The R-L deconvolution always produces non-negative gray level values and works better than linear methods if the blur kernel is known and the noise level in the image is low. Before applying the R-L algorithm, we first estimate

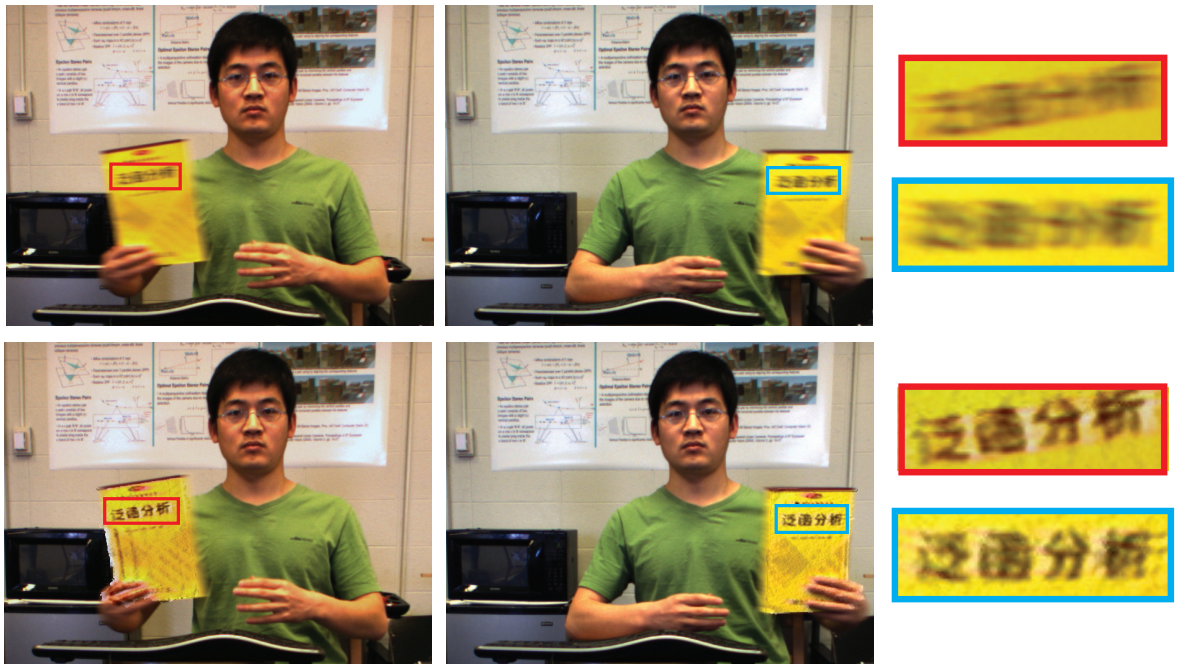


Figure 6.5: Motion deblurring results in a dynamic scene. The yellow book was quickly passed over from one hand to the other. The top row shows two original HR-C frames and the bottom row shows the deblurred results using our approach. The right column shows the closeup of the book. (The subject depicted here is the author of this thesis.)

the masks for each of the foreground regions in the HR-C image that are obtained by using the technique presented in Sec. 6.2.1 and 6.2.2. These masks are used to composite deblurred results into a common background image. In Figure 6.5 and 6.6, we show the motion results on captured dynamic scenes.

6.3 Depth Map Super-resolution

Next, we show how to use the deblurred HR-C image for generating super-resolution depth maps.

6.3.1 Initial Depth Estimation

We first use the two HS-C cameras and apply the graph-cut algorithm [61] to estimate a low-resolution disparity map with respect to the left HS-C camera. We then convert it to

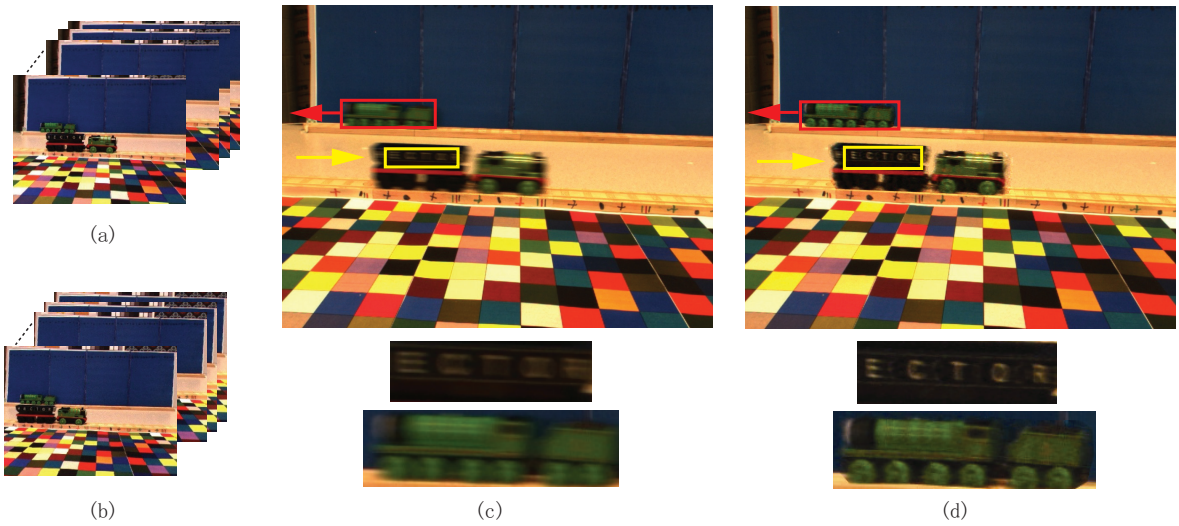


Figure 6.6: Motion deblurring results with spatially varying kernels. (a) and (b) show the two HS-C sequences. (c) shows the HR-C image with strong motion blur. (d) shows the motion deblurred results using our method. Notice the front and back train are moving in opposite directions at different speeds. Our method is able to motion deblur both objects.

the depth map D_L. Next, we warp D_L to the downsampled HR-C camera as $D_0^{'}$. We adopt a similar notation as in [145]: every pixel p_1 in D_L is mapped to its corresponding pixel p_0 in $D_0^{'}$ as:

$$p_0 = (\mathbf{A}_0\ \mathbf{R}_0)\ (\mathbf{A}_1\ \mathbf{R}_1)^{-1}\ (p_1 - \mathbf{A}_1\ \mathbf{T}_1) + \mathbf{A}_0\ \mathbf{T}_0 \tag{6.2}$$

where p is homogeneous coordinate of the pixel as $(sx, sy, s)^T$, s represents the depth of the point, $\mathbf{A}_1$ and $\mathbf{A}_0$ are the camera intrinsic matrices of the HS-C camera S_1 and the downsampled HR-C camera. $\mathbf{R}$ and $\mathbf{T}$ are the extrinsic to the cameras. Here we downsample the HR-C to have the same spatial resolution as HS-C camera to reduce the missing data (holes) in the reprojected depth map.

A sample depth map D_L is shown in Figure 6.7(b). Notice the depth map is incomplete because of the large baseline between the two HS-C cameras. This is less problematic since we focus on capturing the depth map of the moving foreground objects. However, the foreground can exhibit some serious artifacts. For example, in Figure 6.8, the left boundary of the red foreground object in the depth map partially merges with the background and

the silhouettes of the toy train lack fine details. In Sec.6.3.2, we show how to use the joint bilateral upsampling to reduce these artifacts.

We choose to warp the depth map to the HR-C camera instead of warping the HR-C image to the HS-C camera mainly because the depth-based warping inevitably corrupts the quality of the image by introducing holes near occlusion boundaries. Since our goal is to construct a high resolution depth map, it is more preferable to maintain the quality of the high resolution image than the low resolution depth map.

6.3.2 Joint Bilateral Upsampling

To enhance the warped depth map, we use the recently proposed joint bilateral filters. The basic bilateral filter is an edge-preserving filter [117] that uses both a spatial filter kernel and a range filter kernel evaluated on the data values themselves. A joint bilateral filter chooses the range filter from a second guidance image. It can effectively combine flash/no-flash pairs [29, 87], upsample the low resolution exposure maps, and enhance the low resolution range maps [62, 136]. In this paper, we also use the joint bilateral filter for depth map super-resolution.

Our joint bilateral filter combines the low resolution depth map $D_0^{'}$ and the motion deblurred high resolution image $\tilde{I}_0$. It computes the value at each pixel p in the high resolution depth map D_0 as:

$$D_0(p) = \frac{1}{W_p} \sum_{q \in \Theta \backslash \Gamma} G_s(\| p - q \|) G_r(| \tilde{I}_0(p) - \tilde{I}_0(q) |) D_0^{'}(q) \tag{6.3}$$

$$W_p = \sum_{q \in \Theta \backslash \Gamma} G_s(\| p - q \|) G_r(| \tilde{I}_0(p) - \tilde{I}_0(q) |)$$

where G_s is the spatial kernel centered over p and G_r is the range kernel centered at the image value at p in $\tilde{I}_0$; Θ is the spatial support of the kernel G_s. W_p is the normalization factor. Since the warped low resolution depth map $D_0^{'}$ contains holes, we exclude the points Γ that correspond to the holes.

To emphasize on the color difference in the range kernel, we choose

$$| \tilde{I}_0(p) - \tilde{I}_0(q) | = \max(| r_p - r_q |, | g_p - g_q |, | b_p - b_q |) \tag{6.4}$$

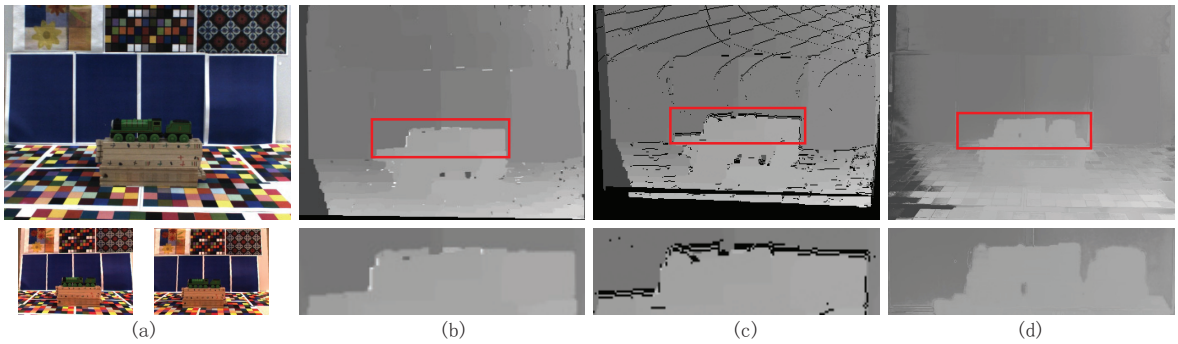


Figure 6.7: Depth map super-resolution results for a static scene. (a) shows the input images from the HS-C and HR-C cameras. (b) shows the depth map from the HS-C camera pair. The result is upsampled using bilinear interpolation for comparison. (c) shows the reprojected low resolution depth map onto the HR-C camera. The warped map is very noisy and contains holes. (d) shows the upsampled results using the joint bilateral filters.

where r, g, b are the color channel of the pixel in $\tilde{I}_0$. In Figure 6.7(c), we show the reprojected low resolution depth map of resolution 320x240 and the enhanced high resolution depth map in Figure 6.7(d). The joint bilateral filter significantly improves the quality of the depth map by filling in the missing holes and partially correcting the inaccurate boundaries (e.g., the background between the engine and carriage of the toy train in Figure 6.7). However, the resulting boundaries still appear blurry due to the spatial Gaussian. Although we can re-apply the joint bilateral filters on the upsampled depth image to further improve the boundary, we implement a simple scheme based on the observation that with a large enough spatial support Θ, there should be enough pixels that have the similar color and depth to pixel p. To do so, we apply a joint median filter: we first pick N pixels from Θ that have the closest color to p and then apply the median filter on the depth values of these N pixels. In our experiment, we choose N to be 20% of all pixels in Θ.

6.4 Results and Discussion

In Figure 6.5 and 6.6, we show the motion deblurring results using our hybrid camera. In Figure 6.5(a), we apply motion deblurring algorithm to a dynamic scene: the yellow book was quickly passed over from the right hand to the left hand. Notice that the texts on the book

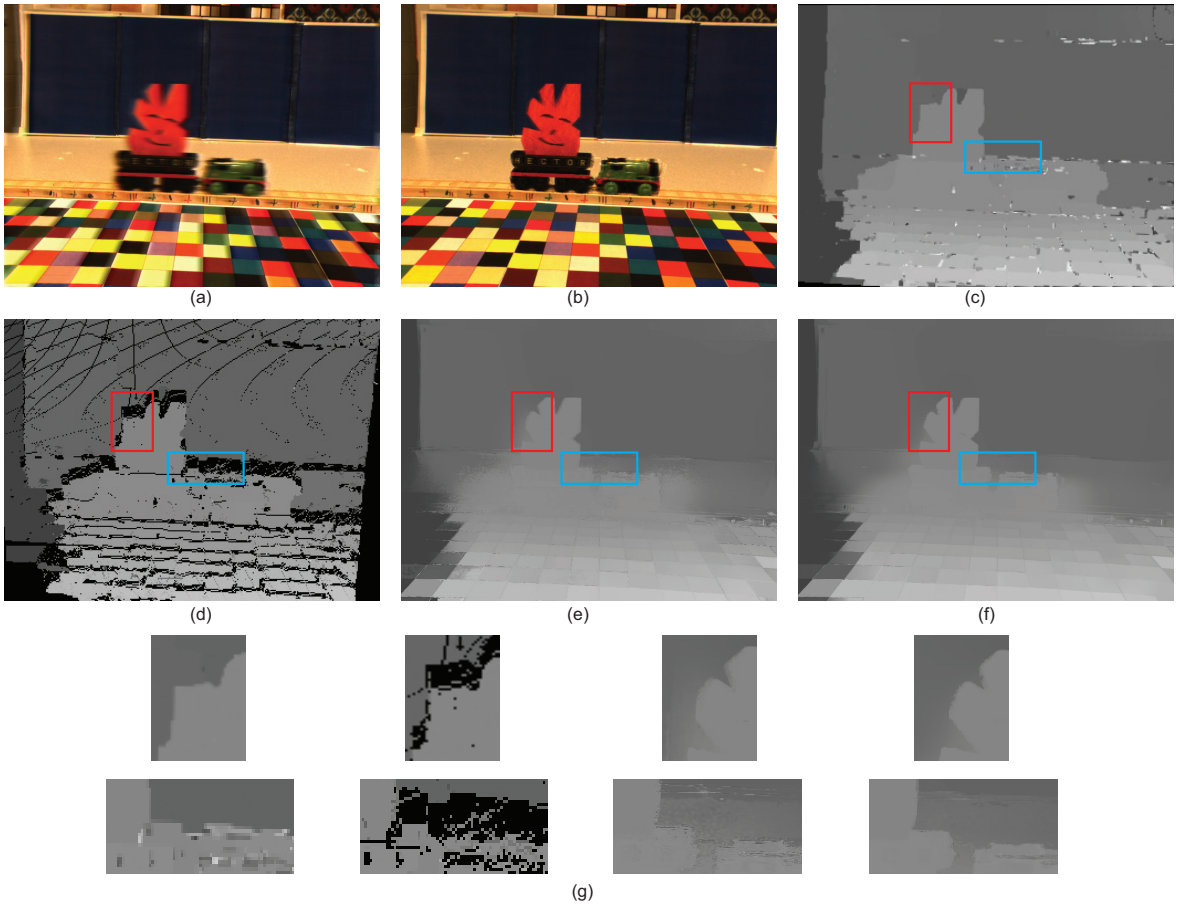


Figure 6.8: Motion deblurring and depth map super-resolution using the hybrid camera. (a) shows the motion blurred image from the HR-C camera. (b) shows the motion deblurred result using the method described in Sec. 6.2. (c) shows the low resolution depth map. We upsample the map using bilinear interpolation to compare it with the bilateral filtered results. (d) is the reprojected depth map. (e) shows the upsampling result using joint bilateral filters. (f) shows further improved results using the joint median filter described in Sec. 6.3.2. (g) shows closeup images of the depth maps.

are unrecognizable in the original HR-C frames, but are clearly readable in the deblurred results shown in Figure 6.5(b). The hand holding the textbook is also effectively deblurred. In Figure 6.6, we use the hybrid camera to deblur spatially varying kernels. The foreground and the background toy trains are moving in opposite directions at different speeds. Since our method separately estimates the PSF for each individual train using the method discussed in Sec. 6.2, we are able to effectively deblur both objects.

In Figure 6.7, we validate our joint bilateral filter method for depth map super resolution in a static scene. We position a toy train on top of the rails. Notice the toy train has complex contours. If we simply upsample the low resolution depth map obtained from the HS-C cameras, the details of the contours disappear. Using a joint bilateral upsampling, we are able to recover these fine details and partially correct the erroneous boundaries in the depth map (e.g., the V shaped contour between the engine and the carriage).

Finally, we show using the hybrid camera to simultaneously motion deblur the HR-C image and reconstruct the high resolution depth map. Figure 6.1 shows the results using our system in a toy train scene. We mount a star shaped object on top of the toy train to emphasize the accuracy of the recovered depth maps. The HS-C and HR-C inputs are shown in Figure 6.1(a) and Figure 6.8(a), and the deblurred result is shown in Figure 6.1(c) and Figure 6.8(b). Our method effectively reduces the motion blur by recovering the text on the toy train and the contour of the star-shaped object, although we also observe some ringing artifacts caused by the R-L deconvolution.

In Figure 6.8, we show the depth super resolution results using joint bilateral filters. We choose a large enough kernel size σ_s for the spatial Gaussian to fill in the missing data. In this example, we set σ_s to be 35. We normalize the depth map to have intensity between $[0\ 1]$ and we set σ_r of the range Gaussian to be 0.02. Figure 6.8(c) shows the depth map from the HS-C pair and Figure 6.8(d) shows the warped result. The reprojected depth map is very noisy and contains many holes. Using the joint bilateral filter, the depth map is significantly enhanced (Figure 6.8(e)). The seams at the occlusion boundaries are further enhanced (Figure 6.8(f)) using the joint median filter as described in Sec.6.3.2. Closeups are shown at the bottom of Figure 6.8. The typical upsampling rate in our experiments is 9. Due

to the camera calibration requirement, we did not further downsample the HS-C camera to demonstrate a higher upsampling rate, although it can easily be achieved if we use a different HR-C camera with a higher resolution.

6.5 Conclusion

In this chapter, we have demonstrated a hybrid camera system for multi-speed image deblurring. Our method estimates the motion flow in the HS-C pair and then warps the flow field to the HR-C camera to estimate the PSF. We then deblurred the HR-C image and used the resulting image to enhance the low-resolution depth map using joint bilateral filters. For image artifacts presented in the deblurred image, such as ringing and saturation, we can easily remove them by adopting the techniques presented in recent literatures about image deblurring [102, 112, 113, 133, 142], or modify the traditional Richard-Lucy deconvolution component to reduce these artifacts.

Besides of motion deblurring and depth map super-resolution, our simplified hybrid camera system still has many other applications in computer vision and graphics. For example, in the absence of motion blurs, it is possible to further simplify our hybrid camera system for dynamic depth of field effect rendering. For example, Yu et al. [140] constructed a simple hybrid stereo camera which consists of one high-resolution color camera and one low-resolution gray-scale camera. They first upsampled the depth map estimated using GPU and then synthesized a light field for rendering dynamic depth of field effects. When compared with Kinect camera system, our hybrid system can work for out-door applications and doesn't have limitations for working space. It is also possible to utilize our hybrid camera system for real-time tracking of blury/non-blury objects. We can assume that the tracked object has smoothy disparity changes between consecutive frames. This can be viewed as the disparity locality, similar as spatial locality which is commonly used in traditional tracking algorithm. In the next chapter, we will discuss how to integrate our hybrid camera and all the algorithms presented in Chap.4-6 for low-light imaging.

Chapter 7

SYSTEM INTEGRATION

From Chapters 3–6, we have elaborated the design of our hybrid camera system and presented three companion computational photography algorithms to use our system for capturing low noise, low motion blur, high quality color imagery under low-light conditions. In each chapter, we have shown experimental results of each individual algorithm, i.e., multi-focus fusion (Chapter 4), multispectral denoising (Chapter 5), and multi-speed deblurring (Chapter 6). In this chapter, we demonstrate putting all three algorithms together for achieving high quality low light imaging.

7.1 Acquiring the Raw Imagery Data

Recall that our hybrid camera aims to combine the advantages of different types of sensors (with respect to aperture, shutter, resolution, and spectrum). Our current setup consists of two Pointgrey Grasshopper high speed monochrome (HS-M) cameras ($640 \times 480 \times 8bit$@120fps), two Pointgrey Flea2 high resolution monochrome (HR-M) cameras ($1024 \times 768 \times 8bit$@30fps), and one single Flea2 high resolution color (HR-C) camera ($1024 \times 768 \times 24bit$@7.5fps). We choose the camera settings so that for each color frame captured at the HR-C camera, we will capture 4 frames on each of the HR-M cameras using a wide aperture, and 16 frames on each of the HS-M cameras using a fast shutter and small aperture.

Fig.7.2 and Fig.7.7(a) show some sample images captured by our system in an indoor setup: a toy training quickly moves towards the left in darkness. Because of the short exposure time, the HS-M images exhibit extremely low contrast. We thus preprocess them using our exposure fusion technique to enhance the contrast. As shown in the top of Fig.7.2, our technique is able to significantly boost the contrast on HS-M frames although the resulting

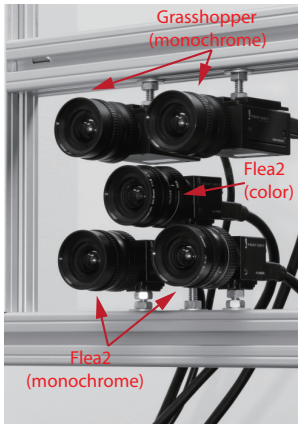


Figure 7.1: Our proposed hybrid camera system.

images still appear rather noisy and cannot be directly used, e.g., for recovering the motion of the toy train. At the bottom of Fig.7.2, we show two sample images captured from the left HR-M camera. By using the wide aperture, the HR-M frames are able to capture images with nearly no noise. In Fig.7.7(a), we further show the corresponding frame in the HR-C camera with respect to Fig.7.2. Notice that the brightness level of this color image is much lower than that of the HR-M images. This is because camera manufacturers install an IR-cut filter and a Bayer pattern before the image sensor to faithfully capture color. These filters, however, not only block NIR and UV light but also decrease the amount of light captured by the sensor by roughly two thirds. The use of long exposure for the HR-C camera can only partially reduce the noise at the expense of introducing motion blurs, as shown in Fig.7.7(a).

7.2 Post-Processing

Next we show how to use our companion algorithms to reconstruct high quality color images from these inputs at 1024x768x24bit@7.5fps. In Chap.4-6, we have developed multi-focus fusion, multi-spectral denoising, and multi-speed motion deblurring modules for reconstructing high quality images under low-light conditions. Fig.7.3 reiterates the pipeline for our processing framework. The main idea is to use HR-M images as the spatial prior to denoise HS-M and HR-C images, and estimate motion information from the denoised HS-M

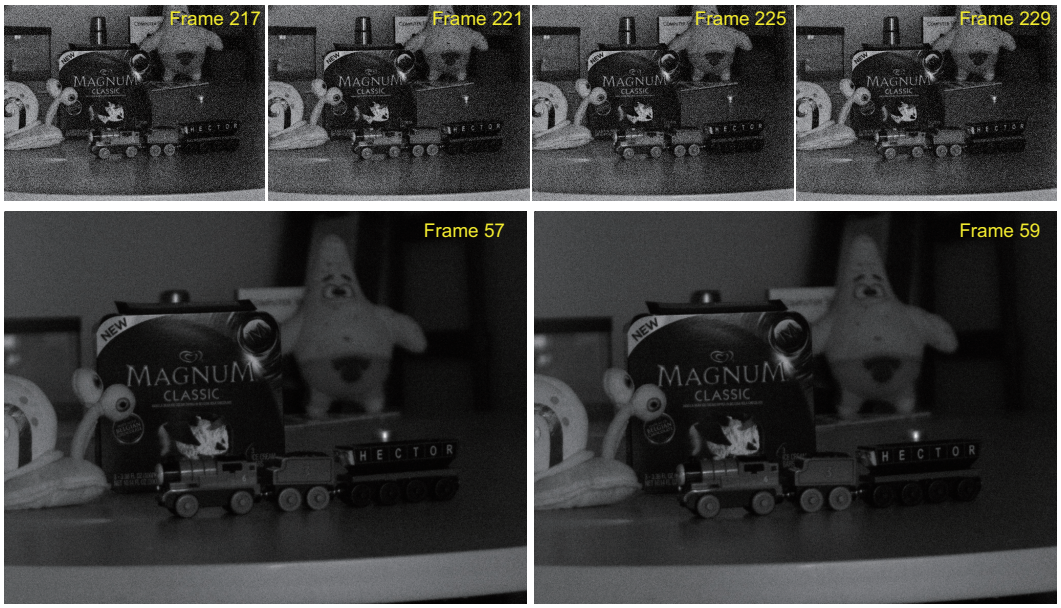


Figure 7.2: An example of images captured under low-light conditions. (a) an image sequence captured from the left HS-M camera. Because the captured images exhibit extremely low contrast due to the use of fast shutters, we show the exposure fused results for instead. (b) an image sequence captured from the left HR-M camera.

sequences for non-blind deconvolution of HR-C images.

Due to the difference in brightness and noise levels and strong parallax between HR-M and HS-M (or HR-C) images, it would be difficult to directly locate corresponding patches as spatial priors from HR-M images for any given patch in HS-M (or HR-C) images. We tackle this problem by first preprocessing the HS-M images to improve the intensity level and then apply BM3D denoising algorithm to partially remove sensor noise. Specifically, we present a novel virtual exposure fusion technique to boost the intensity level of the dark regions while avoiding the bright regions saturating.

Next, we find corresponding patches from HR-M images for any given patch in HS-M images through multi-view block matching, and use it as the spatial prior to regularize the denoising problem together with a ℓ_1 TV term. This spatial term can provide high frequency details which are corrupted by image downsampling and sensor noise at HS-M cameras. The TV regularization can remove unwanted details while preserving important details such as

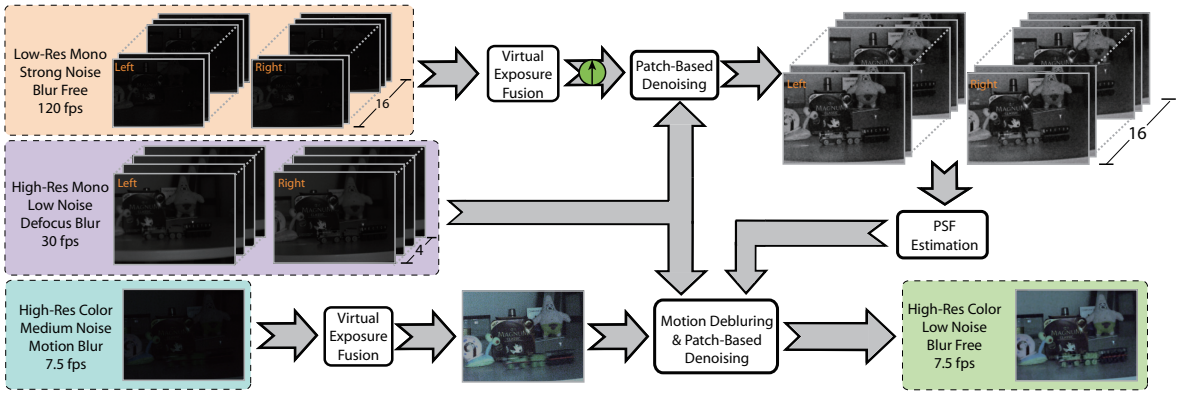


Figure 7.3: System pipeline for low-light imaging. We first preprocess the HS-M images to improve the brightness of dark regions, and then apply BM3D denoising algorithm to partially remove sensor noise. Next we find low-noise patch priors from HR-M images for multispectral denoising of HS-M images. Finally, we design an alternating minimization algorithm to denoise the preprocessed HS-M images.

edges. These terms both regularize the objective function in this minimization problem to effectively reconstruct low-noise HS-M image sequences.

Finally, we motion deblur the HR-C images. We use the denoised HS-M sequences to robustly estimate blur kernels for HR-C images. Because the captured HR-C images contain sensor noise, it is not practical to directly recover latent HR-C images through conventional Richardson-Lucy method used in Chapter 6. Conceptually, we could denoise the HR-C images before we deconvolve them. The denoising process, however, would change the blurred contents and lead to strong visual artifacts in deconvolution as it violates our spatially invariant PSF assumption. Therefore, we choose to use the optimization method presented in [63, 123] to simultaneously remove the blur and noise for fast moving objects. We then apply the same patch-based denoising method presented in Chapter 5 to reconstruct latent images from the input HR-C images.

7.3 Results and Discussions

We evaluate our algorithms on both real and synthetic image sequences. These experiments demonstrate our algorithms are reliable to recover high quality color images under

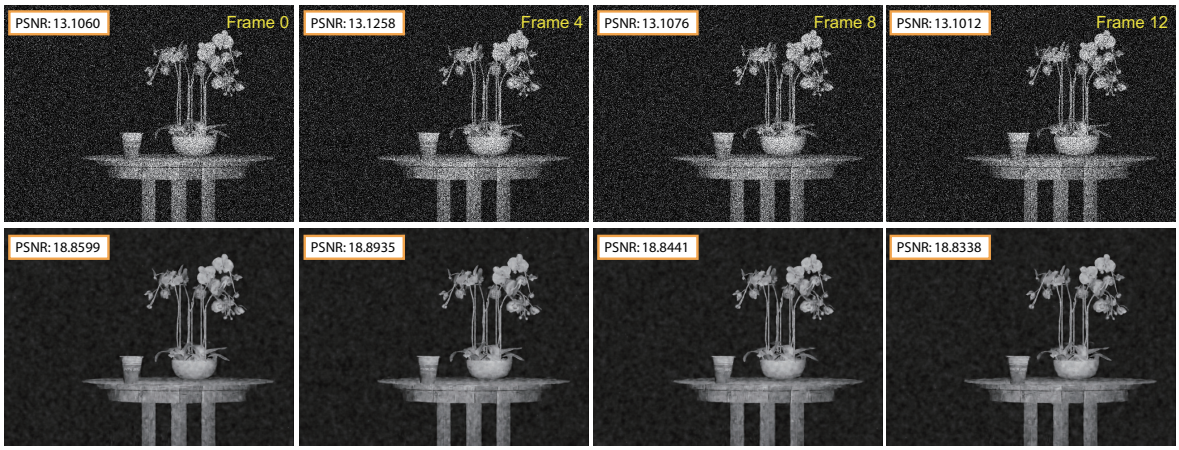


Figure 7.4: Multispectral denoising of a synthetic HS-M image sequence. (top) the synthetic image sequence with zero-mean Gaussian noise of variance 0.09. (bottom) our denoised results.

low-light conditions.

Synthetic Scenes. We use the same synthetic scene shown in Sec. 5.6 to quantitatively demonstrate the performance of our deblurring/denoising algorithms. In Fig.7.4, we show the multispectral denoising of one synthetic HS-M image sequence. Our algorithm improves the PSNR by roughly 5.7 dB over the noisy inputs. Moreover, it can effectively recover the fine details of the degraded HS-M images, for example, the texture of the front table, the characters on the paper cup, and the shape of the background orchid. Unlike the BM3D results, our denoised image sequences only exhibit limited flickering, jumping and other temporal artifacts. This is because the data fidelity term in the objective function of Eq.(5.5) keeps our denoised results close to the original sequence, thus helps to maintain temporal coherence, which is very vital for us to accurately estimate the optical flows from the denoised HS-M sequences.

In Fig.7.5, we show the image deblurring/denoising of a synthetic HR-C image. As discussed in Sec.7.2, we first estimate the motion information of the paper cup from the denoised HS-M image sequences, and then compute the PSF for non-blind deconvolution of the HR-C image, followed by our hybrid denoising method to remove strong sensor noise. Our result not only can reveal the vivid color of the orchid, but also reconstruct the fine

Figure 7.5: Image deblurring/denoising of a synthetic HR-C image. (left) shows the synthesized HR-C image. (center) our deblurring/denoising result. (right) shows the zoomed-in figures corresponding to the rectangle regions in (left) and (center) respectively.

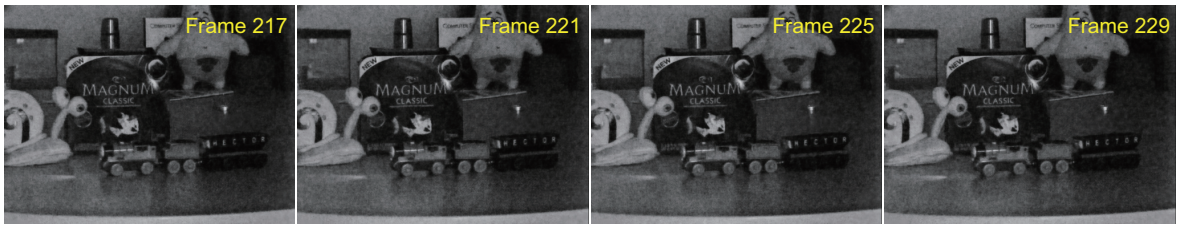


Figure 7.6: Denoising results of corresponding HS-M images shown in Fig.7.2.

details of the fast moving cup, i.e., the design pattern on the cup. In addition, our algorithm impressively improves the PSNR from 17.69 dB to 22.95 dB for the input HR-C image. Here we run the deblurring/denoising algorithm on each color channel separately, and combine the results together to form the final color image.

Real Scenes. Fig.7.6 demonstrates our multispectral denoising results on the contrast-enhanced HS-M images. Compared to the top row of Fig.7.2, not only are strong noise removed, but also fine details, such as the title of the background book, the logo on the chocolate box, and axles of the foreground train, can be clearly recovered in our denoised sequences through the regularization of our multispectral prior and ℓ_1 TV. Next, we warp the estimated optical flows of the moving toy train onto the HR-C camera to estimate the motion blur kernel for image non-blind deconvolution. Notice that the contour and detailed color of the toy train is hardly identifiable in the original HR-C image due to strong noise and motion

(a) (b)

Figure 7.7: Image deblurring/denoising of an input HR-C image. (a) shows the exposure enhanced HR-C image corresponding to HS-M and HR-M sequences in Fig.7.2. (b) our deblurred/denoising result. The bottom row shows the zoomed-in figures corresponding to the rectangle region in the top respectively.

blurs, but are clearly recovered in our result, as shown in Fig.7.7(b). Specifically, our method can even reconstruct the axles of the toy train. Finally, we create an alpha matte (Chapter 6) to combine the deblurred moving train with the background.

To faithfully recover the color information of the deblurred HR-C image, we apply our optimization based denoising algorithm (Chapter 5) to reduce the strong sensor noise. Here we apply our multi-view block matching algorithm only in the green channel, instead of running it for three times. This is due to the fact that a typical Bayer array contains twice as many green as red or blue sensors. The redundancy of green pixels produces a color image with less noise in the green channel after image demosaicing, thus it gives us better patch matching results. As can be seen in Fig.7.7(b), our TV and spatial prior regularized denoising method is able to effectively remove strong sensor noise from the original color image, and keep reconstructed image sharp as well, for example, the characters on the background book and the eyes of the foreground snail.

In Fig.7.8, we apply our new imaging system for acquiring an outdoor scene: a person

Figure 7.8: Multispectral denoising of an outdoor image sequence. (top) the exposure enhanced results of an original HS-M image sequence. (bottom) our denoised results. (The subject depicted here is the author of this thesis.)

Figure 7.9: Motion deblurring of an outdoor HR-C image. (a) an original HR-C image captured under low-light conditions. (b) zoomed-in figure of the rectangle region shown in (a). Estimated blur kernel is displayed in the top left corner. (c) motion deblurring result of (b). (The subject depicted here is the author of this thesis.)

is walking towards the camera array at a road construction site when an SUV passes by. One HR-M camera focuses on the foreground person, and the other one focuses on the background SUV. These cameras are equipped with large aperture lenses to capture low level of noise images, whereas the HS-M cameras are able to capture the fast passing SUV without any blurs. These HS-M images, however, exhibit extremely low contrast due to the use of fast shutters. We first preprocess the input HS-M images through exposure fusion, as shown in the top of Fig.7.8, and then apply our denoising scheme to further reduce the noise using spatial guidance from the HR-M images. Our solution is able to not only remove strong noises on the construction site but also add useful details to the foreground (e.g., the characters on the person's T-shirt) as shown in the bottom of Fig.7.8.

Under some extreme circumstances, it can still be difficult to deblur the HR-C image even after we accurately estimate the blur kernel using the denoised HS-M sequences. For instance, the image of the fast moving SUV (Fig.7.9(a)) exhibit ultra wide dynamic range: the headlights of the SUV are well saturated, while the body of the car is extremely under-exposed and exhibit strong noise. Further, the SUV is moving ultra fast and generates large blur kernels that cannot be robustly handled by most existing deblurring methods including ours. However, our system is still able to denoise the HS-M images well and provide useful information about the vehicle's motion. As shown in the top left of Fig.7.9(b), the accuracy of our estimated PSF can be well measured by the motion of the headlights. Other highlight lines within the figure are reflections of street lamps at the SUV's specular surface.

In summary, we have conducted a number of indoor and outdoor experiments to validate both our system and our algorithms. We have demonstrated putting all three algorithms (multi-focus fusion, multispectral denoising, and multi-speed deblurring) together for achieving high quality low light imaging. Experiments show that our denoised HS-M image sequences have good temporal consistence and are able to provide accurate motion fields of the fast moving objects for deblurring of HR-C images. We have also shown that the reconstructed color image can provide faithful color information of the scene with high resolution.

Chapter 8

CONCLUSION AND FUTURE WORK

In this dissertation, we have presented a new computational imaging system called the hybrid camera array that is suitable for conducting surveillance tasks under low light conditions. Our hybrid camera array leverages recent advances on high speed, high resolution and multi-spectral sensors by integrating multiple types of sensors into a unified imaging system. We have also developed a class of companion computational photography algorithms for fusing the imagery from different types of sensors to produce high resolution, low-noise, low motion blur color video sequences.

8.1 Conclusion

On the hardware side, we have constructed the imaging system (Chapter 3) based on the light field camera array system. We have built a 3×3 light field camera array that is suitable for acquiring 3D dynamic fluid surfaces. We have discussed important issues that affect the design of camera array systems, including bandwidth concern, data recording device, camera synchronization, computer architecture, and etc. The main advantage of our system compared with existing light field camera arrays is that it requires low maintenance and is highly portable and readjustable. To extend the light field array to the hybrid camera array, we integrate a pair of high-resolution monochrome (HR-M) cameras, a pair of high-speed monochrome (HS-M) cameras, and a single high-resolution RGB color (HR-C) camera into a single system. The HR-M cameras are equipped with large aperture lens to gather more light for capturing low-noise images, but with strong defocus blurs. The HS-M cameras capture fast motions without motion blurs but their images are usually very noisy due to fast shutters, and the HR-C camera provides reliable color information of the scene using slow shutters at the cost of strong motion blurs for the fast moving objects.

On the algorithm side, we have developed a class of multi-fusion techniques based on recent breakthroughs in computational photography algorithms for denoising, defocusing and deblurring.

In Chapter 4, we have developed a new multi-focus fusion technique called the Dual Focus Stereo Imaging or DFSI. DFSI uses a pair of images captured from different viewpoints, and at different focuses but with identical wide aperture size. It resembles the HR-M camera pair in our hybrid camera array. Each image in the HR-M camera exhibits different defocus blur and the two images form a defocused stereo pair. To model defocus blur, we have introduced a defocus kernel map (DKM) that computes the size of the blur disk at each pixel. We have also derived a novel *disparity defocus constraint* for computing the DKM in dual focus stereo pair, and integrated DKM estimation with disparity map estimation to simultaneously recover both maps. We have shown that the recovered DKMs provide useful guidance for automatically segmenting and fusing the in-focus regions to create a nearly all-focus image.

In Chapter 5, we have presented a new framework for denoising HS-M images. Our proposed algorithm explores the correlation between the HS-M and HR-M cameras in our hybrid camera array in both spatial and temporal domains. Specifically, we extract in-focus regions from each HR-M image using our DFSI technique (Chapter 4) and then use these low-noise patches as the multispectral priors for HS-M image denoising. We first preprocess the captured HS-M images by using our exposure fusion technique to boost their brightness and contrast. Next, we design an novel optimization scheme and regularize the objective function with a ℓ_1 total variation (TV) term and a multispectral gradient prior. This TV regularization can remove unwanted detail from the noisy image while preserving important details such as edges. Our novel multispectral prior can add in fine details missed due to image noise and low-resolution CCD sensor. We solve this optimization problem efficiently through newly proposed alternating algorithms [114, 123].

In Chapter 6, we have developed a deblurring scheme for reducing motion blurs in the HR-C stream. Our method estimates the motion flow in the denoised HS-M image sequences and then warps the flow field to the HR-C camera to estimate the blur kernel. We have applied

the traditional Richard-Lucy deconvolution algorithm to deblur the HR-C images. Since both the HR-M and the HS-M cameras can provide a depth map of the scene, we have also studied how to use final fused HR-C image to synthesize a high-resolution depth map using the joint bilateral filters.

Finally, we have conducted a number of synthetic and real scene experiments to validate both our system and our algorithms. In Chapter 7, we first design an indoor toy train scene to demonstrate how to put all three algorithms (multi-focus fusion, multispectral denoising, and multi-speed deblurring) together for achieving high quality low light imaging. Experiments show that our denoised HS-M image sequences have good temporal consistence and are able to provide accurate motion fields of the fast moving objects for deblurring of HR-C images. We have also shown that the reconstructed color image can provide faithful color information of the scene with high resolution. At last, we demonstrate some limitations of motion deblurring under extreme conditions, for example, the over saturated headlight motions in an outdoor scene, however, our system is still able to denoise the HS-M image sequences well and provide useful information about the object's motion.

8.2 Future Work

Our proposed hybrid camera system and its computational photography algorithms have the potential to benefit and advance computer vision applications such as generating contents for 3D TV, high speed high resolution imaging, low-light surveillance, and etc.

8.2.1 Capturing Videos for 3D TV

In future, we plan to apply our camera systems for capturing 3D video contents for TVs and movies. As discussed in Chapter 3, our camera systems can support many different 3D video formats, including stereoscopic streams, multi-view streams, and 2D-plus-depth streams, which make our systems ready for many types of 3D displays. For instance, our 3×3 light field camera array can simultaneously provide 9 view video contents for recent lenticular lens 3D TVs (auto-stereoscopic displays), such that they don't need to estimate accurate depth maps for 9 view interpolation. In addition, it is also possible to use our hybrid

camera system (Fig.3.2) to capture 3D video contents, especially for fast moving objects. One advantage of our imaging systems is that the baseline/parallax is adjustable in either software or physical ways. To reduce visual fatigue in 3D TV, we can always postprocess the video contents to adjust the baseline through virtual view synthesis.

8.2.2 High Speed High Resolution Imaging

One important future work that we plan to explore is to generating high-speed and high-resolution videos. Recall that our imaging system is equipped with both high speed and high resolution videos cameras. We hence can potentially further increase the speed and the resolution of the sensors. First, we could employ the time-divided multiplexing technique discussed in Sec.3.1 to improve the speed of these video streams by a factor of two. Next, we could design a dictionary learning based image super-resolution method to super-resolve HS-M stereo videos. Recent studies [129, 130, 135] have shown that dictionary based on image patches exhibit nice sparsity properties and can use techniques similar to compressive sensing to achieve superresolution. Different from [135], image patches that constitute the dictionary could be directly extracted from our HR-M image sequences, in a similar way as our denoising method. At last, to generate color for the imagery, we can adopt color transfer techniques [70, 96, 126] to transfer color from HR-C images to superresolved high speed grayscale stereo video streams.

8.2.3 Using Temporal Coherence

It is important to note that our denoising technique does not consider temporal coherence between image frames, thus the denoised HS-M sequences suffer from temporal artifacts, such as flickering and jumping, even though the data fidelity term in the optimization framework reduces these artifacts to some extent. An intuitive way to improve our results is to extend our multi-view block matching method to explore not only spatial neighbors across different view points but also their temporal neighbors. We can define time windows to bound the block matching in temporal domain, based on our camera synchronization scheme.

It is also possible to adopt the virtual exposure framework [10] to use temporal integration for video denoising of HS-M images.

Temporal information can benefit motion deblurring as well. Agrawala et al. [2] have shown that motion blur in successive video frames can be invertible by changing the exposure time of successive frames. Therefore, another important part of future work is to explore motion deblurring in temporal domain by considering consecutive HR-C frames. To solve motion debluring of saturated regions, we could also design a new deconvolution framework which can directly utilize the low-resolution object patch sequences besides the estimated object motions/PSFs.

8.2.4 Real-time Implementation

In Chapter 5, we have demonstrated the performance gains achieved by exploring the data parallelism using the OpenMP API. Many other components of our low-light imaging system could also be implemented using OpenMP to reduce the running time, e.g, optical flow estimation and image deconvolution. With latest advances in graphics hardware, we may further improve the performance our algorithm by migrating them onto the Graphics Processing Unit (GPU). Recall that almost all our processes on the imagery data exhibit strong data parallelism. Therefore, we can use NVidia's CUDA architecture and tools make this data parallel computing on a GPU every straightforward. Specifically, disparity map (and defocus kernel map) estimation, frequently used in Chapters 4, 5 and 6, could be solved on the GPU by adopting the methods described in [41]. We can also improve the computational efficiency of blur kernel size estimation (described in Chapter 4), and multi-view block matching (Chapter 5).

8.2.5 Potential Extensions

Application in Face Recognition. Compared to other night vision technologies, such as thermal imaging and image intensifiers, our imaging system can generate images with better quality under low-light conditions, and even with reliable color information, which makes our hybrid camera array suitable for biometrics security systems. When coupled with

NIR illuminators, we can adopt similar approaches proposed in [47,75] to robustly recognize human faces at night, even under extreme weather conditions, such as, fog and mist. Another thing to mention is that we can facilitate object detection at night by considering different reflectance of NIR for various objects. Together with estimated depth maps of HS-M images, they could help us prune out large regions for object detection.

Application in Tracking. Visual tracking plays an important role in robotics, human computer interaction, medical imaging and many other computer vision applications. Tremendous efforts have been focused on separately handling noise, illumination, occlusions, blurs, and background clutter. As discussed in the previous chapters, our hybrid camera system is able to reduce image noise, remove motion blurs and improve illumination, which can makes tracking methods more robust on these challenging image inputs. Moreover, since our camera system has large parallax in both x and y direction on the image plane, it can handle object occlusion as well. At last, multiple instances of the same object at different view points can benefit the image feature extraction as well, besides the estimated HS-M depth maps and object motion fields. For details about object tracking, please refer to Yilmaz et al. [137].

BIBLIOGRAPHY

[1] Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen. Interactive digital photomontage. *ACM Trans. Graph.*, 23(3), 2004.

[2] A. Agrawal, Y. Xu, and R. Raskar. Invertible motion blur in video. *ACM Transactions on Graphics (TOG)*, 28(3):1–8, 2009.

[3] Narendra Ahuja and A. Lynn Abbott. Active stereo: Integrating disparity, vergence, focus, aperture and calibration for surface estimation. *Pattern Analysis and Machine Intelligence*, pages 1007–1029, Oct. 1993.

[4] Simon Baker and Iain Matthews. Lucas-kanade 20 years on: A unifying framework. *Int. J. Comput. Vision*, 56(3):221–255, 2004.

[5] Yosuke Bando, Bing-Yu Chen, and Tomoyuki Nishita. Extracting depth and matte using a color-filtered aperture. *ACM Trans. Graph.*, 27(5), 2008.

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[7] M. Ben-Ezra and S.K. Nayar. Motion deblurring using hybrid imaging. *Computer Vision and Pattern Recognition, 2003.*, 1:I–657–I–664 vol.1, 18-20 June 2003.

[8] M. Ben-Ezra and S.K. Nayar. Motion-based motion deblurring. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(6):689–698, June 2004.

[9] Eric P. Bennett, J. L. Mason, and Leonard McMillan. Multispectral bilateral video fusion. *IEEE Transactions on Image Processing*, 16(5):1185–1194, 2007.

[10] Eric P. Bennett and Leonard McMillan. Video enhancement using per-pixel virtual exposures. *ACM Trans. Graph.*, 24(3):845–852, 2005.

[11] James R. Bergen, P. Anandan, Keith J. Hanna, and Rajesh Hingorani. Hierarchical model-based motion estimation. *ECCV '92*, pages 237–252, 1992.

[12] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23, 2001.

[13] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient n-d image segmentation. *Int. J. Comput. Vision*, 70(2):109–131, 2006.

[14] A. Buades and B. Coll. A non-local algorithm for image denoising. In *CVPR '05*, pages 60–65, 2005.

[15] P. Burt and E. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on communications*, 31(4):532–540, 1983.

[16] E.J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *Information Theory, IEEE Transactions on*, 52(2):489–509, 2006.

[17] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

[18] Canon. What is optical image stabilizer? *http://www.canon.com/bctv/faq/optis.html*, 2006.

[19] T.F. Chan and C.K. Wong. Total variation blind deconvolution. *Image Processing, IEEE Transactions on*, 7(3):370–375, 1998.

[20] P. Chatterjee and P. Milanfar. Clustering-based denoising with locally learned dictionaries. *Image Processing, IEEE Transactions on*, 18(7):1438–1451, 2009.

[21] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *Image Processing, IEEE Transactions on*, 16(8):2080–2095, 2007.

[22] Shengyang Dai and Ying Wu. Removing partial blur in a single image. *Computer Vision and Pattern Recognition, 2009*, pages 2544–2551, 2009.

[23] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes*, SIGGRAPH '08, pages 31:1–31:10, New York, NY, USA, 2008. ACM.

[24] Y. Ding, J. Yu, and P. Sturm. Multiperspective stereo matching and volumetric reconstruction. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1827–1834. IEEE, 2009.

[25] Yuanyuan Ding, Feng Li, Yu Ji, and Jingyi Yu. Dynamic fluid surface acquisition using a camera array. In *ICCV '11*, 2011.

[26] W. Dong, X. Li, L. Zhang, and G. Shi. Sparsity-based image denoising via dictionary learning and structural clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[27] Frédo Durand and Julie Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 257–266, 2002.

[28] Jr. EARL HAMMON. Practical post-process depth of field. *GPU Gems 3*, 28:583–606, 2007.

[29] Elmar Eisemann and Frédo Durand. photography enhancement via intrinsic relighting. *ACM Trans. Graph.*, 23(3):673–678, 2004.

[30] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, 15(12):3736–3745, 2006.

[31] Hany Farid and Eero P. Simoncelli. Range estimation by optical differentiation. *J. Optical Society of America*, 15:1777–1786, 1998.

[32] Raanan Fattal, Dani Lischinski, and Michael Werman. Gradient domain high dynamic range compression. *ACM Trans. Graph.*, 21:249–256, July 2002.

[33] P.F Felzenszwalb and D.R. Huttenlocher. Efficient belief propagation for early vision. pages 261–268, 2004.

[34] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman. Removing camera shake from a single photograph. *ACM Trans. Graph.*, 25(3):787–794, 2006.

[35] Christian Frese and Ioana Gheta. Robust depth estimation by fusion of stereo and focus series acquired with a camera array. *Multisensor Fusion and Integration for Intelligent Systems*, pages 243–248, 2006.

[36] D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Transactions on pattern analysis and machine intelligence*, pages 367–383, 1992.

[37] D. Geman and C. Yang. Nonlinear image recovery with half-quadratic regularization. *Image Processing, IEEE Transactions on*, 4(7):932–946, 1995.

[38] T. Georgiev, C. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala. Spatio-angular resolution tradeoffs in integral photography. In *Eurographics Symposium on Rendering*, pages 263–272. Citeseer, 2006.

[39] T.G. Georgiev and A. Lumsdaine. Superresolution with plenoptic 2.0 cameras. In *Signal Recovery and Synthesis*. Optical Society of America, 2009.

[40] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Sciences*, 2:323, 2009.

[41] S. Grauer-Gray, C. Kambhamettu, and K. Palaniappan. Gpu implementation of belief propagation using cuda for cloud tracking and reconstruction. In *Pattern Recognition in Remote Sensing (PRRS 2008), 2008 IAPR Workshop on*, pages 1–4. IEEE.

[42] Paul Green, Wenyang Sun, Wojciech Matusik, and Frédo Durand. Multi-aperture photography. In *SIGGRAPH '07*, page 68, 2007.

[43] R. M. Haralick, S. R. Sternberg, and X. Zhuang. Image analysis using mathematical morphology. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(4):532–550, 1987.

[44] Samuel W. Hasinoff and Kiriakos N. Kutulakos. Light-efficient photography. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, 2008.

[45] Samuel W. Hasinoff and Kiriakos N. Kutulakos. confocal stereo. *Int. J. Comput. Vision*, 81(1):82–104, 2009.

[46] M.R. Hestenes. Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320, 1969.

[47] D. Huang, Y. Wang, and Y. Wang. A robust method for near infrared face recognition based on extended local binary pattern. In *Proceedings of the 3rd international conference on Advances in visual computing-Volume Part II*, pages 437–446. Springer-Verlag, 2007.

[48] Adrian Ilie and Greg Welch. Ensuring color consistency across multiple cameras. In *ICCV '05*, pages 1268–1275, Washington, DC, USA, 2005. IEEE Computer Society.

[49] Aaron Isaksen, Leonard McMillan, and Steven J. Gortler. Dynamically reparameterized light fields. In *SIGGRAPH '00*, pages 297–306, 2000.

[50] J. Jia, J. Sun, C.K. Tang, and H.Y. Shum. Bayesian correction of image intensity with spatial consideration. *ECCV*, 3:342–354, 2004.

[51] Jiaya Jia. Single image motion deblurring using transparency. In *Computer Vision and Pattern Recognition*, June 2007.

[52] N. Joshi, S.B. Kang, C.L. Zitnick, and R. Szeliski. Image deblurring using inertial measurement sensors. *ACM Transactions on Graphics (TOG)*, 29(4):30, 2010.

[53] N. Joshi, C.L. Zitnick, R. Szeliski, and D.J. Kriegman. Image deblurring and denoising using color priors. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1550–1557. IEEE, 2009.

[54] Neel Joshi, Wojciech Matusik, and Shai Avidan. Natural video matting using camera arrays. *ACM Trans. Graph.*, 25(3):779–786, 2006.

[55] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, 1988.

[56] Jangheon Kim and Thomas Sikora. Confocal disparity estimation and recovery of pinhole image for real-aperture stereo camera systems. *ICIP*, pages 229–232, 2007.

[57] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pages 15–18. IEEE, 2006.

[58] Y. Kojima, R. Sagawa, T. Echigo, and Y. Yagi. Calibration and performance evaluation of omnidirectional sensor with compound spherical mirrors. In *Proc. The 6th Workshop on Omnidirectional Vision, Camera Networks and Non-classical cameras*. Citeseer, 2005.

[59] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *ICCV*, pages 508–515, 2001.

[60] Vladimir Kolmogorov and Ramin Zabih. Multi-camera scene reconstruction via graph cuts. In *ECCV*, pages 82–96, 2002.

[61] Vladimir Kolmogorov and Ramin Zabih. Multi-camera scene reconstruction via graph cuts. *European Conference on Computer Vision*, May 2002.

[62] Johannes Kopf, Michael F. Cohen, Dani Lischinski, and Matt Uyttendaele. Joint bilateral upsampling. *SIGGRAPH '07*, page 96, 2007.

[63] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. *Advances in Neural Information Processing Systems*, 22:1–9, 2009.

[64] Dilip Krishnan and Rob Fergus. Dark flash photography. *SIGGRAPH '09*, 2009.

[65] D. Lanman, D. Crispell, M. Wachs, and G. Taubin. Spherical catadioptric arrays: Construction, multi-view geometry, and calibration. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pages 81–88. IEEE, 2006.

[66] Sungkil Lee, Gerard Jounghyun Kim, and Seungmoon Choi. Real-time depth-of-field rendering using anisotropically filtered mipmap interpolation. *IEEE Transactions on Visualization and Computer Graphics*, 15(3):453–464, 2009.

[67] H. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H.P. Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Transactions on Graphics (TOG)*, 22(2):234–257, 2003.

[68] A. Levin. Blind motion deblurring using image statistics. *Advances in Neural Information Processing Systems*, 2006.

[69] A. Levin, R. Fergus, F. Durand, and W.T. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics (TOG)*, 26(3), 2007.

[70] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 689–694. ACM, 2004.

[71] A. Levin and B. Nadler. Natural image denoising: Optimality and inherent bounds. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[72] A. Levin, Y. Weiss, F. Durand, and W.T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1964–1971. Ieee, 2009.

[73] M. Levoy, B. Chen, V. Vaish, M. Horowitz, I. McDowall, and M. Bolas. Synthetic aperture confocal imaging. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 825–834. ACM, 2004.

[74] Marc Levoy and Pat Hanrahan. Light field rendering. In *SIGGRAPH '96*, pages 31–42, 1996.

[75] S.Z. Li, RuFeng Chu, ShengCai Liao, and Lun Zhang. Illumination invariant face recognition using near-infrared images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):627–639, April 2007.

[76] Yuanzhen Li, Lavanya Sharan, and Edward H. Adelson. Compressing and companding high dynamic range images with subband architectures. *ACM Trans. Graph.*, 24:836–844, July 2005.

[77] C.K. Liang, T.H. Lin, B.Y. Wong, C. Liu, and H.H. Chen. Programmable aperture photography: multiplexed light field acquisition. In *ACM SIGGRAPH 2008 papers*, pages 1–10. ACM, 2008.

[78] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[79] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 689–696. ACM, 2009.

[80] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *ICCV*, pages 2272–2279, 2009.

[81] Morgan McGuire, Wojciech Matusik, Hanspeter Pfister, John F. Hughes, and Frédo Durand. Defocus video matting. In *SIGGRAPH '05*, pages 567–576, 2005.

[82] T. Mertens, J. Kautz, and F. Van Reeth. Exposure fusion. In *Computer Graphics and Applications, 2007. PG'07. 15th Pacific Conference on*, pages 382–390. IEEE, 2007.

[83] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, and Pat Hanrahan. Light field photography with a hand-held plenoptic camera. In *Tech. Rep. CSTR 2005-02, Stanford University Computer Science, Apr.*, 2005.

[84] Nikon. Precise camera-shake compensation at every angle. *http://www.nikon.co.jp/main/eng/portfolio/about/technology/nikon_technology/vr_e/index.htm*, 2005.

[85] Sylvain Paris and Fredo Durand. A fast approximation of the bilateral filter using a signal processing approach. *ECCV 2006*.

[86] Alex P. Pentland. A new sense for depth of field. *PAMI*, 9(4):523–531, 1987.

[87] Georg Petschnigg, Richard Szeliski, Maneesh Agrawala, Michael Cohen, Hugues Hoppe, and Kentaro Toyama. Digital photography with flash and no-flash image pairs. *ACM Trans. Graph.*, pages 664–672, 2004.

[88] Marc Pollefeys, Reinhard Koch, and Luc Van Gool. A simple and efficient rectification method for general motion. *ICCV '09*.

[89] Michael Potmesil and Indranil Chakravarty. A lens and aperture camera model for synthetic image generation. In *SIGGRAPH '81*, pages 297–305, 1981.

[90] Matan Protter and Michael Elad. Image sequence denoising via sparse and redundant representations. *Trans. Img. Proc.*, 18(1):27–35, 2009.

[91] Ashish Raj and Ramin Zabih. A graph cut algorithm for generalized image deconvolution. *ICCV '05*, pages 1048–1054, 2005.

[92] A.N. Rajagopalan, S. Chaudhuri, and Uma Mudenagudi. Depth estimation and image restoration using defocused stereo pairs. *Pattern Analysis and Machine Intelligence*, 26(11):1521–1525, Nov. 2004.

[93] Ramesh Raskar, Amit Agrawal, and Jack Tumblin. Coded exposure photography: motion deblurring using fluttered shutter. *ACM Trans. Graph.*, 25(3), 2006.

[94] Ramesh Raskar, Kar-Han Tan, Rogerio Feris, Jingyi Yu, and Matthew Turk. Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. *ACM Trans. Graph.*, 23(3):679–688, 2004.

[95] Ramesh Raskar, Jack Tumblin, Ankit Mohan, Amit Agrawal, and Yuanzen Li. Computational photography. In *Proc. Eurographics STAR*, 2006.

[96] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley. Color transfer between images. *Computer Graphics and Applications, IEEE*, 21(5):34–41, 2001.

[97] Erik Reinhard, Michael Stark, Peter Shirley, and James Ferwerda. Photographic tone reproduction for digital images. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '02, pages 267–276, New York, NY, USA, 2002. ACM.

[98] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec. *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2006.

[99] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, 2004.

[100] LI Rudin and S. Osher. Total variation based image restoration with free local constraints. In *ICIP*, pages 31–35, 1994.

[101] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

[102] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *ACM Trans. Graph.*, 27(3), 2008.

[103] Qi Shan, Jiaya Jia, Sing Bing Kang, and Zenglu Qin. Using optical defocus to denoise. *Computer Vision and Pattern Recognition*, 2010.

[104] Qi Shan, Wei Xiong, and Jiaya Jia. Rotational motion deblurring of a rigid object from a single image. In *ICCV*, Oct. 2007.

[105] R. Sibson. *A brief description of natural neighbour interpolation*. In Vic Barnet, editor, Interpreting Multivariate Data. John Wiley & Sons, Chichester, 1981.

[106] C.V. Stewart. Robust parameter estimation in computer vision. *Siam Review*, 41(3):513–537, 1999.

[107] D. Sun, S. Roth, and M.J. Black. Secrets of optical flow estimation and their principles. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2432–2439. IEEE, 2010.

[108] D. Sun, E.B. Sudderth, and M.J. Black. Layered image motion with explicit occlusions, temporal consistency, and depth ordering. *Advances in Neural Information Processing Systems*, 2010.

[109] Jian Sun, Yin Li, Sing Bing Kang, and Heung-Yeung Shum. Symmetric stereo matching for occlusion handling. In *CVPR*, pages 399–406, 2005.

[110] Jian Sun, Heung-Yeung Shum, and Nan-Ning Zheng. Stereo matching using belief propagation. In *PAMI*, volume 25, 2003.

[111] Y. Taguchi, A. Agrawal, A. Veeraraghavan, S. Ramalingam, and R. Raskar. Axial-cones: modeling spherical catadioptric cameras for wide-angle light field rendering. In *ACM Transactions on Graphics (TOG)*, volume 29, page 172. ACM, 2010.

[112] Yu-Wing Tai, Hao Du, Michael S. Brown, and Stephen Lin. Correction of spatially varying image and video motion blur using a hybrid camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1012–1028, 2010.

[113] Y.W. Tai, P.Tan, L.Gao, and M.S. Brown. Richardson-lucy deblurring for scenes under projective motion path. Technical report, KAIST, 2009.

[114] M. Tao and J. Yang. Alternating direction algorithm for total variation deconvolution in image reconstruction. *Preprint*, 2009.

[115] M.F. Tappen, B.C.Russell, and W.T. Freeman. Efficient graphical models for processing images. *Computer Vision and Pattern Recognition*, 2:II–673–II–680 Vol.2, 2004.

[116] A.N. Tikhonov, V.Y. Arsenin, and F. John. *Solutions of ill-posed problems*. Vh Winston Washington, DC, 1977.

[117] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, page 839, 1998.

[118] D.L. Tull and A.K. Katsaggelos. Iterative restoration of fast-moving objects in dynamic image sequences. *Optical Engineering*, 35(12):3460–3469, December 1996.

[119] J. Unger, A. Wenger, T. Hawkins, A. Gardner, and P. Debevec. Capturing and rendering with incident light fields. In *Proceedings of the 14th Eurographics workshop on Rendering*, pages 141–149. Eurographics Association, 2003.

[120] V. Vaish, M. Levoy, R. Szeliski, C.L. Zitnick, and S.B. Kang. Reconstructing occluded surfaces using synthetic apertures: Stereo, focus and robust measures. 2006.

[121] Ashok Veeraraghavan, Ramesh Raskar, Amit Agrawal, Ankit Mohan, and Jack Tumblin. Dappled photography: mask enhanced cameras for heterodyned light fields and coded aperture refocusing. In *SIGGRAPH '07*, page 69, 2007.

[122] Jue Wang and M.F. Cohen. Optimized color sampling for robust matting. In *CVPR*, 2007.

[123] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272, 2008.

[124] Z.F. Wang and Z.G. Zheng. A region based stereo matching algorithm using cooperative optimization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Ieee, 2008.

[125] Masahiro Watanabe and Shree K. Nayar. Rational filters for passive depth from defocus. *Int. J. Comput. Vision*, 27(3):203–225, 1998.

[126] T. Welsh, M. Ashikhmin, and K. Mueller. Transferring color to greyscale images. In *ACM Transactions on Graphics (TOG)*, volume 21, pages 277–280. ACM, 2002.

[127] B. Wilburn, N. Joshi, V. Vaish, M. Levoy, and M. Horowitz. High-speed videography using a dense camera array. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–294. IEEE, 2004.

[128] B. Wilburn, N. Joshi, V. Vaish, E.V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. *ACM Transactions on Graphics (TOG)*, 24(3):765–776, 2005.

[129] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T.S. Huang, and S. Yan. Sparse representation for computer vision and pattern recognition. *Proceedings of the IEEE*, 98(6):1031–1044, 2010.

[130] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 210–227, 2008.

[131] Yalin Xiong and Steven A. Shafer. Depth from focusing and defocusing. In *CVPR*, pages 68–73, 1993.

[132] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010.

[133] Li Xu and Jiaya Jia. Two-phase kernel estimation for robust motion deblurring. *ECCV '10*, 2010.

[134] J.C. Yang, M. Everett, C. Buehler, and L. McMillan. A real-time distributed light field camera. In *Proceedings of the 13th Eurographics workshop on Rendering*, pages 77–86. Eurographics Association, 2002.

[135] Jianchao Yang, John Wright, Thomas Huang, and Yi Ma. Image super-resolution as sparse representation of raw image patches. *Computer Vision and Pattern Recognition*, 2008.

[136] Qingxiong Yang, Ruigang Yang, James Davis, and David Nister. Spatial-depth super resolution for range images. *Computer Vision and Pattern Recognition*, June 2007.

[137] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *Acm Computing Surveys (CSUR)*, 38(4):13, 2006.

[138] Y. Yitzhaky, I. Mor, A. Lantzman, and N. S. Kopeika. Direct method for restoration of motion-blurred images. *Journal of the Optical Society of America A: Optics, Image Science, and Vision*, 15(6):1512–1519, 1998.

[139] J. Yu and L. McMillan. General linear cameras. *Computer Vision-ECCV 2004*, pages 14–27, 2004.

[140] Z. Yu, C. Thorpe, X. Yu, S. Grauer-Gray, F. Li, and J. Yu. Dynamic depth of field on live video streams: A stereo solution. *Computer Graphics International*, 2011.

[141] Lu Yuan, Jian Sun, Long Quan, and Heung-Yeung Shum. Image deblurring with blurred/noisy image pairs. *SIGGRAPH '07*, page 1, 2007.

[142] Lu Yuan, Jian Sun, Long Quan, and Heung-Yeung Shum. Progressive inter-scale and intra-scale non-blind image deconvolution. *ACM Trans. Graph.*, 27(3), 2008.

[143] Li Zhang and Shree Nayar. Projection defocus analysis for scene capture and image display. *ACM Trans. Graph.*, 25(3):907–915, 2006.

[144] Li Zhang, S. Vaddadi, Hailin Jin, and S.K. Nayar. Multiple view image denoising. *Computer Vision and Pattern Recognition*, pages 1542–1549, 2009.

[145] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, 2000.

[146] Changyin Zhou and Shree Nayar. What are Good Apertures for Defocus Deblurring? In *IEEE International Conference on Computational Photography*, Apr 2009.

[147] C. Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High-quality video view interpolation using a layered representation. *ACM Trans. Graph.*, 23(3):600–608, 2004.

Appendix

PERMISSION LETTER

The following is the permission letter from Mr. Xuan Yu for using his pictures in this dissertation.

Sub: Permission Letter

To whom it may concern:

I'm writing to let you know that I am permitting my labmate Feng Li to use my portrait pictures in his dissertation, paper, and other forms of publications, for research purposes only. I do not, however, give permission for any other use or for any re-disclosure of this information.

Yours Faithfully,

Xuan Yu

Xuan Yu 11/30/2011