Point Spread Function Engineering for Scene Recovery

Changyin Zhou

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2012

ABSTRACT

Point Spread Function Engineering for Scene Recovery

Changyin Zhou

A computational camera uses a combination of optics and processing to produce images that cannot be captured with traditional cameras. Over the last decade, a range of computational cameras have been proposed, which use various optics designs to encode and using computation to decode useful visual information. What is often missing, however, is the quantitative connection between camera design and the captured visual information, and little systematic work has been done to evaluate and optimize these computational camera designs. While computational cameras can be designed in complicated ways, many of them can be effectively characterized by their point spread functions (PSFs): the intensity distribution on an image sensor as a response to a point light source in a scene.

This thesis explores the techniques to characterize, evaluate and optimize computational cameras via PSF engineering for various scene recovery tasks in computer vision. I first demonstrate the quantitative connection between PSF and the loss of image detail in blurry images. A captured image can appear blurry for a number of reasons, including defocus, lens aberration, atmospheric turbulence, and object motion. Image blurring can be formulated as a convolution of the latent sharp image and a PSF, and deconvolution techniques must be used to recover details from a blurred region. Here, I propose a comprehensive framework of PSF evaluation for the purpose of image deblurring, which takes the effects of image noise, deblurring algorithm, and the structure of natural images into account.

In the case of defocus blur, it is well known that the shape of a defocus PSF is largely determined by the aperture pattern of the camera lens. By using the derived evaluation criterion, it is possible to optimize the pattern of lens aperture to preserve many more image details when defocus occurs. Both through simulations and experiments, I demonstrate the significant improvement gained by using optimized coded apertures.

While defocus causes a loss in image detail, it also encodes depth in images. A typical depth from defocus (DFD) technique computes depth from two images captured with circular apertures of different sizes. Circular apertures produce circular defocus PSFs. In this thesis, I present that the use of a circular aperture severely restricts the accuracy of DFD, and propose a comprehensive framework of PSF evaluation for depth recovery. With this framework, we can derive a criterion for evaluating a pair of apertures with respect to the precision of depth recovery. This criterion is optimized using a genetic algorithm and gradient descent search to arrive at a pair of high resolution apertures. The two coded apertures are found to complement each other in the scene frequencies they preserve. With this property it becomes possible to not only recover depth with greater fidelity but also to obtain a high quality all-focused image from the two defocused images.

While depth recovery can significantly benefit from optimized aperture patterns, its overall performance is rigidly limited by the lens aperture's physical size. To transcend this limitation, I propose a novel depth recovery technique using an optical diffuser - referred to as depth from diffusion (DFDiff).

I show that DFDiff is analogous to conventional DFD, in which the scatter angle of the diffuser determines the system's effective aperture. High precision depth estimation can be achieved by choosing a proper diffuser and no longer requires the large lenses that DFD requires. Even a consumer camera with a low-end small lens can be used to do high-precision depth estimation when coupled with an optical diffuser. In my detailed analysis of the image formation properties of a DFDiff system, I show a number of examples demonstrating greater precision in depth estimation when using DFDiff.

While the finite depth of field (DOF) of a lens camera leads to defocus blur, it also produces artistic visual experience. Many of today's displays are interactive in nature, which opens up a possibility for new kind of visual representations. Users could, for example, interactively refocus images to different depths, so that they can experience the artistic narrow DOF images while simultaneously making available the image detail for the entire image. To enable image refocusing, one typical approach is to capture the entire light field. But this method has the drawback of a significant sacrifice in spatial resolution due to the dimensionality gap: the captured information (light field) is 4D, while the required information (focal stack) is only 3D.

In this thesis, I present an imaging system that directly captures focal stacks by a sweeping

focal plane. First, I describe how to synchronize focus sweeping with image capturing so that the summed DOF of a focal stack efficiently covers the entire depth range. Then, I take a customized algorithm to enable a seamless refocusing experience, even in textureless regions or with moving objects. Prototype cameras are presented to capture real space-time focal stacks. There is also an interactive refocusing viewer available online at *www.focalsweep.com*.

Table of Contents

List of Figures

List of Tables

Dedicated to my beloved mother, father, brother and sisters,
and to my beloved wife Elaine.

Chapter 1

Introduction

1.1 Point spread function engineering for scene recovery

A camera is a device that collects light (Figure 1.1 (a)). Over the last century, the evolution of cameras has been truly remarkable. Throughout the course of this evolution, however, the basic model underlying cameras has remained essentially unchanged (Figure 1.1 (b)). The traditional camera has a detector and a lens that captures only the principal rays passing through its optical center, and produces the familiar perspective image. In other words, the traditional camera performs a very simple and restrictive sampling of the complete set of rays, or the light field, that resides in real scene [108] [109] [183].

A computational camera (Figure 1.1 (c)) combines novel optics and computation to produce the final image. The novel optics are used to map rays from the scene onto pixels on the detector in an unconventional fashion. For example, the ray shown in Figure 1.1 (c) is geometrically redirected by the optics to a different pixel than the one it would have reached in the case of a traditional camera. As illustrated by the change in color from yellow to red, the ray can also be photometrically altered by the optics. Although the images captured by computational cameras are optically coded and may not be visually meaningful in their raw form, the information can be recovered by using computation. In all cases, the new arrangement of the rays helps to encode more useful visual information in the captured images compared to conventional cameras.

Over the last decade, a wide variety of computational cameras have been developed, which all encode more useful visual information in the captured images as compared to conventional cam-

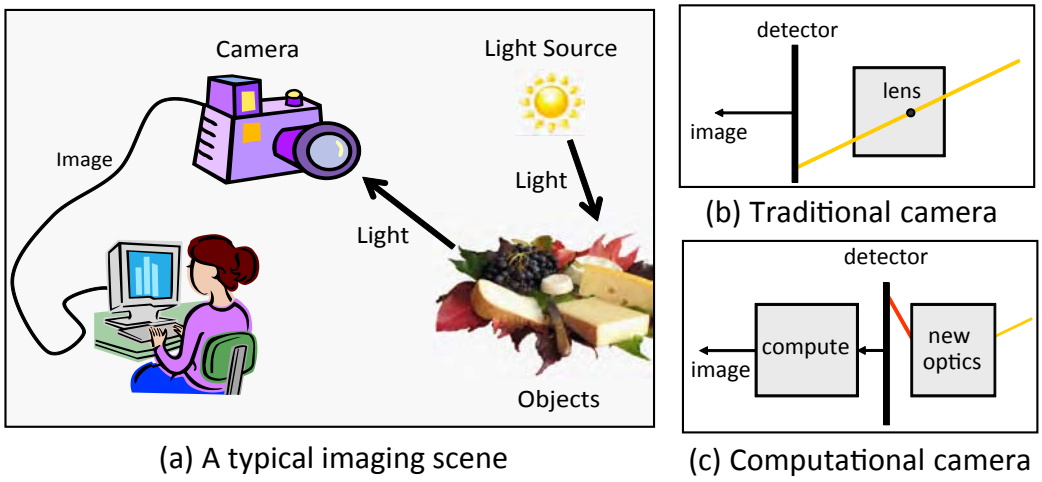


Figure 1.1: (a) In a typical scene for imaging, light rays from sources are reflected by objects, collected by camera lens, and then converted to digital signals for further processing. (b) A traditional camera model captures only those principal rays that pass through its center of projection to produce the familiar linear perspective image. (c) A computational camera uses optical coding followed by computational decoding to produce new types of images.

eras. The coding methods used in today's computational cameras can be broadly classified into six approaches: object side coding, pupil plane coding, sensor side coding, illumination coding, camera arrays and clusters, and unconventional imaging systems ([109] [183]). The design space for the optics of computational cameras is large. It would be desirable to have a single design methodology that produces an optimized optical system for any given set of imaging specifications. This optimization criteria would have to formulate the complex optical systems, which can be pretty complicated, and incorporate a variety of factors, including performance and complexity. However, such a systematic design and optimization approach is largely missing in literature. As a consequence, just as in the case of traditional optics, the design of computational cameras remains part science and part art.

In this thesis, I explore the techniques to optimize computational camera designs using point spread function (PSF) engineering. PSF is the intensity distribution on a camera sensor as a response to a point light source in the scene. It gives an efficient and simple way to characterize imaging systems. The amount of the spreading is often used directly as a measure for the quality of an

imaging system. For a typical Lambertian scene without occlusion, the image formation can be formulated as an integral of the corresponding PSF of every scene point:

$$F(x,y) = \int_{p\in\Omega} K(x,y|p) \cdot I(p) dp,$$

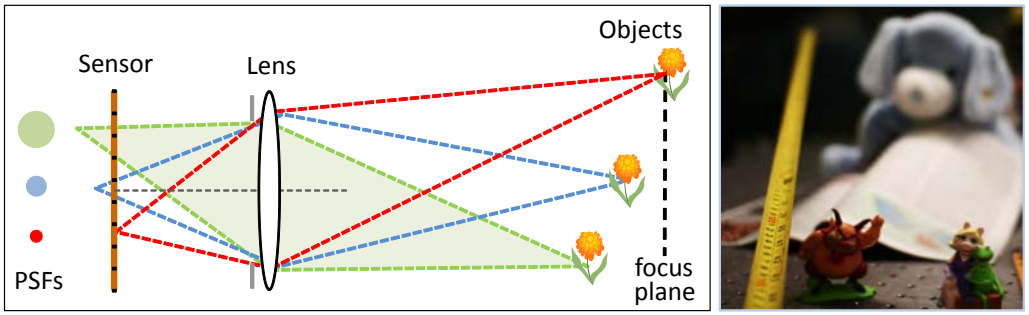
where p is any visible 3D point on the scene geometry Ω, $I(p)$ is the light intensity at p, and $K(x,y|p)$ is the PSF for the point p. This image formation is well known to be a process of dimension reduction, in which a large amount of information is lost. We can see that the PSF $K(x,y|p)$ is the kernel of the mapping from a 3D scene to a 2D image in this process. As the kernel of image formation, PSF determines how the scene texture $I(p)$ is distorted and how the scene geometry Ω is encoded in the 2D image.

This kernel $K(x,y|p)$ is solely determined by camera design. For example, in an ideal pinhole camera model, $K(x,y|p)$ is a Dirac delta function by ignoring diffraction; in a typical thin lens model, $K(x,y|p)$ is a disk function, whose scale is determined by the relative position of p to the focal plane (shown in Figure 1.2); and for most lens designs, the PSFs due to diffraction and lens aberrations can be concisely modeled based on their lens profiles by using Zernike polynomials [179]. By properly designing a computational camera via PSF optimization, I will be able to preserve more useful information for scene recovery.

One fundamental problem is how to precisely model the connection between PSFs and the useful information for scene recovery. Once we have a precise model, we would be able to analytically evaluate any camera design via its PSF, and then accordingly optimize the camera design. In this thesis, we address this problem in the context of various tasks of scene recovery and make the following contributions:

- Texture detail of a scene is often lost in a captured image due to defocus, lens aberration, or diffraction. We study the effects of PSFs in recovering scene texture from blurry images, and propose a close-form criterion to evaluate the "goodness" of PSFs according to the expected quality of deblurring.

  Defocus is the most commonly seen image blur in photographs. For a traditional camera as shown in Figure 1.2, an object will appear in-focus when it is on the focus plane and will appear blurry as it deviates from the focus plane. The shape of defocus PSF is determined



(a) The geometry of defocus and its PSFs (b) An image with defocus blur

Figure 1.2: (a) The geometry of defocus in a traditional lens camera. Objects at greater distances away from the focal plane will appear increasingly blurred. On the left are the defocus PSFs for three distances. (b) A captured image with defocus blur.

by the aperture pattern and its scale is related to object depth. We therefore use the proposed criterion to optimize aperture patterns for defocus deblurring.

- While defocus causes a loss in image detail, it also encodes depth information of the scene. We propose a comprehensive framework of evaluating PSFs for depth recovery, and use it to solve for an optimized pair of coded apertures.

- While aperture coding optimizes the PSF and helps to improve the precision of depth recovery, the sensitivity of depth estimation is rigidly limited by aperture size ([144]). To transcend this fundamental limit, we propose using an optical diffuser to modulate the PSFs and this leads to a novel depth recovery technique – referred to as depth from diffusion (DFDiff).

- The finite depth of field (DOF) of a lens camera leads to defocus blur, but this also produces artistic visual experience. It is an effective tool to draw user attention selectively to a specific part of the scene. In this thesis, we propose capturing a stack of images in a duration when the focus sweeps over a large depth range in the scene – referred to as space-time focal stack. We then design a novel image refocusing algorithm using the space-time focal stack. This allows users to experience the artistic narrow DOF appearance of the scene while simultaneously making available the image detail for the entire image.

1.2 Related work

In this thesis, we study PSF optimization for both recovering image and depth of scenes, and have applied the derived PSF evaluation criteria for lens aperture optimization. We have also proposed using a novel optical device to overcome the limit of DFD, and have designed a focal sweep camera to capture more information for space-time image refocusing. In the past decades, a large number of related work has been done for similar applications in scene recovery.

1.2.1 On image recovery

In the 1960s, coded aperture techniques were introduced in the field of high-energy astronomy as a novel way of PSF engineering. These techniques have been used for improving signal-to-noise ratio for lensless imaging of x-ray and γ-ray sources [1][26]. In subsequent decades, many different aperture patterns were proposed, including the popular multiplexed uniformly redundant array (MURA) [56]. Unfortunately, the coded apertures designed for lensless imaging are not optimal to use within lenses for defocus deblurring, as observed in [166].

Also in the 1960s, researchers in the field of optics began developing unconventional apertures to capture high frequencies with less attenuation. Binary aperture patterns [169] [165] as well as continuous ones [101] [122] were proposed and analyzed in detail. The patterns proposed in the optics community were chosen in an ad-hoc fashion (based on intuitions) and then analyzed in details in terms of their optical transfer functions.

It is only in the last few years that the design of apertures for defocus deblurring has been posed as an optimization problem. In particular, Veeraraghavan et al. [166] performed gradient descent search to improve the MURA pattern [56] and then binarized the resulting pattern. Due to the large search space associated with the optimization, they restricted themselves to binary patterns with 7×7 cells. The criterion used in [166] maximizes the minimum of the power spectrum of the aperture pattern. In our work, we show that apertures with higher performance can be achieved by taking image noise and image statistics into consideration.

In addition to coded aperture, there are other competing PSF engineering techniques. Wavefront coding method modulates the aperture by using a 3D phase plate. This technique was first introduced by Dowski and Cathey [41] to extended the depth of field. They show analytically that a

camera with a cubic phase plate produces a PSF that is approximately depth invariant and therefore one can recover a focus image by a single deconvolution. Besides, several different designs of phase plates are given in [27] [48] also for extended depth of field. Diffusion coding [36] and focal sweep [70] [106] are other two PSF engineering techniques that can be used to preserve more image detail. Our derived evaluation criterion can also be applied to optimize parameters in focal sweep and cubic phase plate cameras.

1.2.2 On depth recovery

While defocus results in image blur, it also encodes depth information in 2D images. Depth from defocus (DFD) technique has been studied extensively by assuming circular apertures in the past decades (a few samples are [124] [156] [116] [129] [167] [159] [43]). These work either assume the PSFs of an imaging system are pillbox (or cylindrical) functions, or assume they are Gaussian. Partly owing to the good mathematical properties of pillbox or Gaussian functions, people have been able to develop a variety of effective DFD algorithms.

Also, a lot of analysis and optimization on these DFD algorithms and camera settings were conducted based on the assumption of pillbox or Gaussian PSFs. Subbarao and Tyan [157] study how the image noise affects the performance of a spatial-domain DFD approach proposed in [159]. Schechner and Kiryati [143] analyze the effect of focus setting on the DFD method implemented by axially moving the sensor, and reveal the change in focus setting should better be less than twice the depth of field. Rajagopalan and Chaudhuri [129] discuss the effect of degree of relative blurring on the accuracy of the depth estimation and proposed a criterion for optimal selection of camera parameters. Especially, they show that for a Gaussian aperture pair, the optimal radius ratio is 1.73, which is very close to the optimization result in this thesis.

To improve depth estimation, Levin et al. [88] proposed using an aperture pattern with a more distinguishable pattern of zero-crossings in the Fourier domain than that of the conventional circular apertures. Similarly, Dowski [40] designed a phase plate that has responses at only a few frequencies, which makes their system more sensitive to depth variations. These methods specifically target depth estimation from a single image, and rely heavily on specific frequencies and image priors. A consequence of this strong dependence is that they become sensitive to image noise and cannot distinguish between a defocused image of a sharp texture and a focused image of smoothly vary-

ing texture. Moreover, these methods compromise frequency content during image capture, which degrades the quality of image deblurring.

A basic limitation of using a single coded aperture is that aperture patterns with a broadband frequency response are needed for optimal defocus deblurring but are less effective for depth estimation [88], while patterns with zero-crossings in the Fourier domain yield better depth estimation but exhibit a loss of information for deblurring. Since high-precision depth estimation and high-quality defocus deblurring generally cannot be achieved together with a single image, we propose in this thesis addressing this problem by taking two images with different coded apertures optimized to jointly obtain a high-quality depth map and an all-focused image.

Multiple images with different coded apertures were used for DFD in [42] [72]. In [42], two images are taken with two different aperture patterns, one being Gaussian and the other being the derivative of a Gaussian. These patterns are such designed so that depth estimation involves only simple arithmetic operations, making it suitable for real-time implementation. Hiura and Matsuyama [72] aims for more robust DFD by using a pair of pinhole apertures within a multi-focus camera. The use of pinhole pairs facilitates depth measurement. However, this aperture coding is far from optimal. Furthermore, small apertures significantly restrict light flow to the sensor, resulting in considerable image noise that reduces depth accuracy. Long exposures can be used to increase light flow but will result in other problems such as motion blur.

Greengard et al. [58] exploits 3D diffraction effects to make spatially rotating PSFs by using a 3D optical phase plate. The PSF rotates as the depth changes and is used for depth estimation. Hasinoff and Kutulakos [67] propose to capture a large set of images of a scene with predetermined focii and apertures of the lens. From these images, one can reconstruct the scene with high geometric complexity and fine-scale texture.

1.2.3 On image refocusing

Per user click, image refocusing displays a narrow DOF image, in which the clicked pixel appears focused. A typical approach is to capture the entire light field and use the light field to render a stack of narrow DOF images.

The concept of light field has been used for a long history. In the early 20^{th} century, Ives [76], Lippmann [95] have proposed plenotic camera designs to capture light fields. The idea of light

field resurfaced in the community of computer vision and graphic in the late 1990s when Levoy and Hanrahan [91] and Gortler et al. [54] described the 4D parameterization of light fields and show how new views can be rendered by using light field data. A stack of images with different focus can also be rendered from a light field, and then be used for image refocusing.

A number of light field cameras have been designed and made in recent years. Levoy et al. [92] used a plenoptic camera to capture the light field of specimens and propose algorithms to compute a focal stack from a single light field image, which can be processed as in deconvolution microscopy to produce a 3D sharp volume. Ng et al. [119] and Ng [118] use the same plenoptic camera design and emphasizes its application in image refocusing. Georgeiv et al. [49] and Georgiev and Intwala [50] show a number of variants of light field camera designs for different trade-off between spatial and angular resolution. Light field cameras can also be built using camera arrays [170] or coded aperture techniques [93].

Rendering a focal stack from a light field image requires sacrificing spatial resolution significantly. This is because of the dimensionality gap the captured information (light field) is 4D, while the required information (focal stack) is only 3D. A lot of redundant information is captured by light field cameras.

There are other approaches that use an all-in-focus image and a depth map to render a focal stack for image refocusing [88]. These approaches usually involve complicated processes of image rendering. More importantly, they usually assume that scenes are Lambertian and have no occlusion. As a result, their rendered narrow DOF images often suffer severely from image artifacts and look unnatural for scenes with non-Lambertian surface and occlusions. An inaccurate depth map will also lead to errors in image refocusing. In this thesis, we propose a focal sweep camera that captures focal stack directly for image refocusing. By avoiding the dimension gap in capturing and complicated image rendering in processing, this design provides users high-quality full-resolution images at every focus with minimal computation cost.

1.3 Thesis organization

In Chapter 2, I introduce related technical background in point spread function (PSF) and depth of field (DOF), and briefly review the work in the research area of computational camera. Chapter 3

addresses the PSF optimization problem for image deblurring and I use the proposed PSF evaluation criterion to optimize aperture pattern for defocus deblurring. Chapter 4 discusses PSF optimization problem for depth from defocus (DFD). In Chapter 5, I use optical diffuser to module PSFs for high-precision depth recovery. In Chapter 6, I present a focal sweep imaging system that can capture space-time focal stacks for image refocusing. Chapter 7 concludes the thesis.

Chapter 2

Background and overview

2.1 Point spread function and depth of field

2.1.1 Point spread function

Point spread function (PSF) is the response of an imaging system to a point source in a scene. The amount of the spreading is often used as a measure for the quality of an imaging system. In practice, a PSF is often a combination of multiple optical effects, including diffraction, aberration, defocus, veiling glare, and etc.

Figure 2.1 illustrates four of the most typical optical effects and their corresponding PSFs. Diffraction occurs because light as a wave will bend around obstacles and spread past them (a). In a typical lens camera, the spreading of the diffraction PSF is proportional to the wavelength and the lens f-number, which is the ratio of the focal length to the aperture diameter. The shape of diffraction PSF of a circular aperture is often referred to as airy disk, as shown on the right of in Figure 2.1(a).

Optical aberration is a departure in the performance of an optical system from the predictions of paraxial optics [59]. Typical optical aberration includes spherical aberration, coma, astigmatism, chromatic aberration, field of curvature, distortion and other effects. Figure 2.1 illustrates the geometry of spherical aberration (b) and coma (c), and their corresponding PSFs using spot diagrams. To compensate for aberrations, modern lens design applies lenses of different shapes and materials ([77] [149] [47]). Defocus (d) is one particular type of optical aberrations, which occurs when

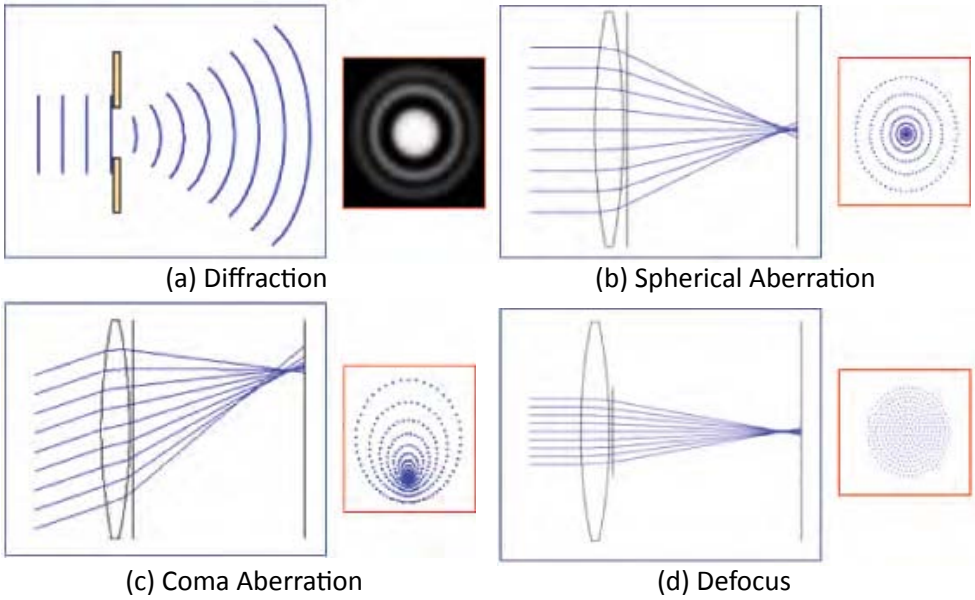


Figure 2.1: Illustrate four of the typical optical phenomena and their resulting PSFs. (a) An illustration of diffraction and its PSF. (b) Geometry of spherical aberration and its PSF (shown as spot diagram). (c) Geometry of coma aberration and its PSF (shown as spot diagram). (d) Geometry of defocus and its PSF (shown as spot diagram). All spot diagrams are simulated by Zemax [47] via ray tracing.

objects are out of focus and is an effect familiar to almost every camera user.

In Fourier optics of incoherent light, the relation between a wave function and its resulting PSF can be simply described by a Fresnel transform [20] [53]:

$$f(x) = |\mathcal{F}(W(x) \cdot Q_d(x))|^2, \tag{2.1}$$

where $f(x)$ is the PSF function, $\mathcal{F}(\cdot)$ is the Fourier transform, $W(x)$ is the wave function at the aperture plane, and $Q_d(x)$ is a quadratic phase term determined by focus distance d. Equation 2.1 holds as long as the f-number is not extremely small and the field angle is not too large [53], and applies to most cameras used in the computer vision and graphics fields. For coherent light, the PSF will simply be $\mathcal{F}(W(x) \cdot Q_d(x))$. While Fourier optics allows an understanding of the wave physics, most discussions in this thesis are in the realm of geometrical optics.

In geometrical optics, light propagation is described in terms of rays and all optical systems therefore become linear. PSF describes how the linear imaging system responds to a point light

source in the scene. In cases in which the PSF is invariant to translation (or location of light source), the imaging system becomes linear and translation-invariant. Therefore, according to the convolution theory, the captured image $i(x)$ can be formulated by a convolution of the latent focused image $i_0(x)$ and the PSF $f(x)$:

$$i(x) = i_0(x) \otimes f(x).$$

In practice, the PSFs of diffraction, defocus, and various lens aberration are not perfectly invariant to translation over the entire depth range and field of view. For example, defocus PSF changes with depth, and lens aberration changes with field angle. However, since they are approximately invariant in local regions, it is still proper to formulate the captured images using convolution. PSF(s) still stand as a concise way to model an optical system.

The convolution theorem states that $\mathcal{F}[f \otimes g] = \mathcal{F}[f] \cdot \mathcal{F}[g]$, where $\mathcal{F}[f]$ denotes the Fourier transform of f. Therefore, a captured image $i(x) = i_0(x) \otimes f(x)$ can be written as

$$I(\xi) = I_0(\xi) \cdot F(\xi)$$

in the Fourier domain. Here we use the upper case letter I, I_0 and F to denote the Fourier transforms of images i, i_0, and f.

The power spectrum of PSF $|F(\xi)|$ is often referred to as Modulation Transfer Function (MTF) and is frequently used to measure the optical quality of imaging systems in optical design.

2.1.2 Depth of field

In a conventional camera, for an image detector at any location, there is one focal plane that is perfectly focused according to the Thin Lens Law. The depth of eld (DOF) is the range between the nearest and farthest objects in a scene that appear acceptably sharp in an image. A lens of circular aperture produces circular PSFs and so the sharpness of an image can be measured by the size of the PSF. The acceptable size (or often referred to as circle of confusion) is influenced by viewing condition, presenting format, and other factors. For digital imaging, a popularly accepted size is the pixel size, or twice the pixel size, if a Bayer color filter array [14] is used with image sensor to produce RGB color images. The letter c denotes the circle of confusion in this thesis.

The DOF of a conventional camera is determined by the focal length f, f-Number N (the ratio of focal length to aperture diameter), and the focus position z. When the focus position z is large in

comparison to f, it can be derived [84] [147]:

$$DOF \approx \frac{2Ncf^2z^2}{f^4 - N^2c^2z^2} \tag{2.2}$$

from the Thin Lens Law. In particular, when $z >= \frac{f^2}{Nc}$, the DOF will cover the infinity, and therefore $\frac{f^2}{Nc}$ is therefore often called hyperfocal distance. It is obvious that DOF is inversely related to the aperture size.

Given a conventional lens camera, there is a fundamental trade-off between DOF and image signal-to-noise ratio (SNR). DOF can be increased by stopping down the aperture. However, this reduces the amount of light received by the sensor, resulting in lower SNR. This trade-off leads to images of lower quality as spatial resolution increases in recent years (or when there is a decrease in the circle of confusion). This is because a smaller circle of confusion c yields a smaller DOF (according to Equation 2.2). At the same time, a smaller pixel collects less light. This trade-off between DOF and SNR is one of the fundamental and long-standing limitations of imaging.

2.2 Computational camera: concept and taxonomy

A computational camera uses a combination of novel optics and computation to produce a final image. Although the images captured by computational cameras are optically coded and may not be visually meaningful in their raw form, the information can be recovered by using computation. This combination of novel optics and computation hence can produce new types of images that are potentially beneficial to a vision system. The coding methods used in today's computational cameras can be broadly classified into six approaches: object side coding, pupil plane coding, illumination coding, camera clusters or arrays, and unconventional coding [109] [183].

2.2.1 Object side coding

Object side coding (Figure 2.2 (a)) attaches external devices to the camera and is probably the most convenient way to implement computational cameras. For the distance between the optical element and the lens, the cones of light rays from objects at different field angles will intersect with the element in different areas. As a result, if the surface profile is not homogeneous, object side coding will yield spatially varying modulation. This property has been widely used to encode more useful visual information and can be found in various applications.

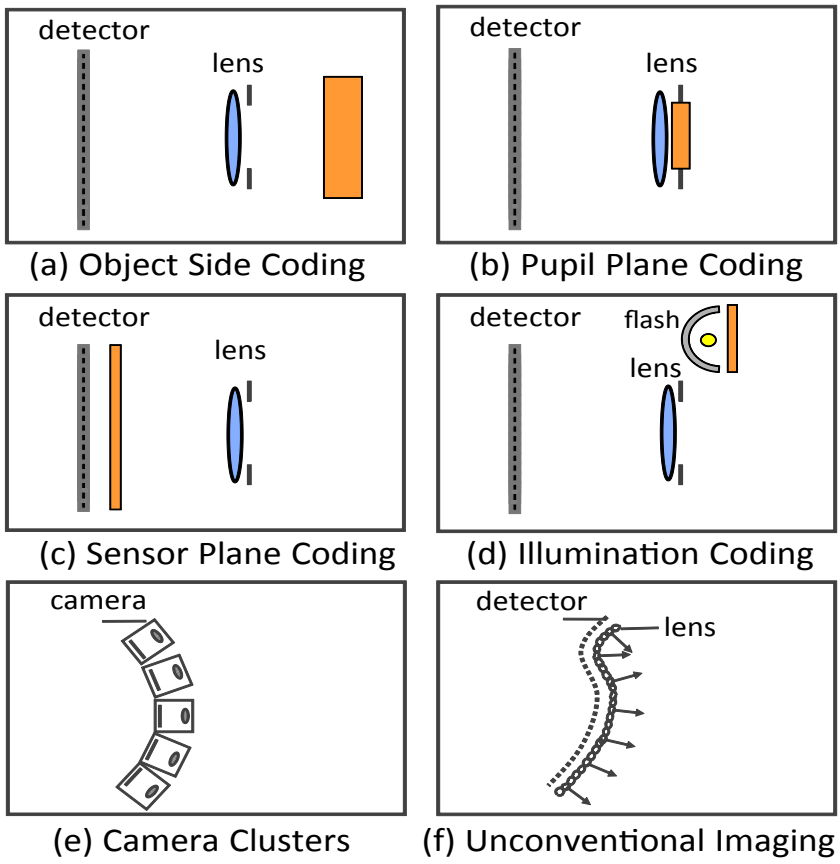


Figure 2.2: Optical coding approaches used in computational cameras. (a) Object side coding, where an optical element is attached externally to a conventional lens. (b) Pupil plane coding, where an optical element is placed at, or close to, the aperture of the lens. (c) Sensor side coding, where an optical element is behind the lens. (d) Imaging systems that make use of coded illumination. (e) Imaging systems that are made up of a cluster or array of traditional camera modules. (f) Imaging systems using unconventional camera architectures or non-optical devices.

Lee et al. [85] proposed using a bi-prism in front of lens for stereo vision with a single camera. Light rays from any single point will be split into two by the bi-prism and produce two image points on the sensor as if viewed from two viewpoints. This yields an effect of stereo. Georgeiv et al. [49] propose using an array of lens-prism pairs in front of the main lens to capture light fields (shown in Figure 2.3 (a)). The information captured by the sensor can be used to reconstruct the 4D light field. In [49] and [50], the authors also mentioned other possible object side configurations for light field acquisition by arranging prisms and lenses in different ways.

Catadioptric techniques combine lenses and mirrors in camera design and are often used to increase camera FOV [19] [23] [29] [32] [73] [176] [81] [79]. These techniques have significant impact on a variety of real-world applications, including surveillance, autonomous navigation, virtual reality, and video conferencing [21] [171] [31].

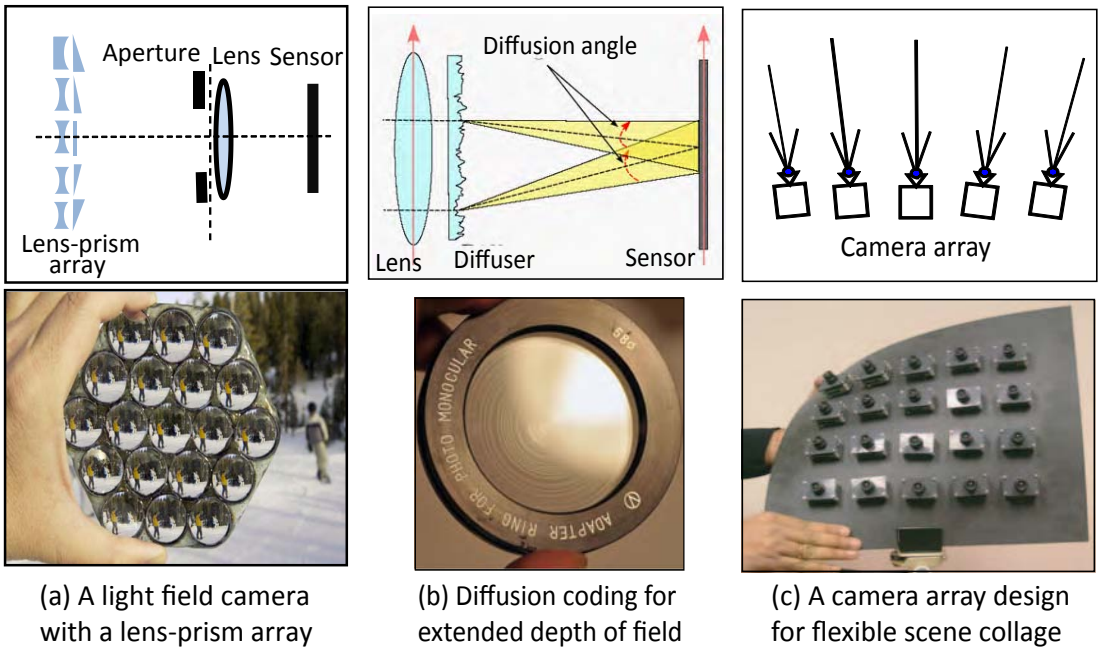
Another type of object side coding, although less common, has been proposed by using homogeneous filters. For example, Umeyama and Godin [163] and Nayar et al. [113] propose capturing images with differing polarization directions in order to remove specular reflections. Rouf et al. [138] use a star filter mounted in front of a cameras to encode the visual information for saturated areas and then use computation to recover high dynamic range images.

2.2.2 Pupil plane coding

Pupil plane coding (Figure 2.2 (b)) places optical elements (or an optical element) at or close to the pupil plane of a traditional lens. Since any rays from objects ideally pass through the same pupil plane, pupil plane coding can be used to provide spatially invariant light modulation and to manipulate the system PSF.

Pupil plane coding using intensity modulators is often referred to as coded aperture techniques or sometimes also apodizer techniques in optics. When diffraction and optical aberration are negligible, the shape of the PSF is simply determined by the aperture pattern, and the scale is determined by the amount of defocus. Previous optics research has proposed using coded apertures (e.g., [169] [122]) to preserve more high frequency information in the case of defocus. In astronomy, optimized patterns such as Modified Uniformly Redundant Array (MURA) are often used for lensless imaging [44] [56] in order to improve the signal-to-noise ratio of the captured images.

Pupil plane coding using phase modulators is often referred to as wavefront coding. A phase



(a) A light field camera with a lens-prism array

(b) Diffusion coding for extended depth of field

(c) A camera array design for flexible scene collage

Figure 2.3: Examples of computational cameras. (a) Object side coding: a light field camera using an array of lens-prism pairs. On the top is the camera geometry; and on the bottom is the lens-prism array. (b) Pupil plane coding: a diffusion coding camera for extended depth of field. On the top is the camera geometry; and on the bottom is a sample of the coded diffuser that is attached to the lens. (c) A camera array designs for flexible scene collage. On the top is the geometry of the design; and on the bottom shows the camera array.

modulator is usually a plate of glass of certain 3D profile. A phase plate will distort the input light field in the angular dimension, and the resulting PSF will simply be the histogram of the derivation of the wavefront function. In wave optics, this relation can be formulated using Equation 2.1. Wavefront coding techniques have been studied for decades in optics for a variety of applications. Dowski [40] designed a phase plate that has responses at only a few frequencies, which makes the imaging system more sensitive to depth variations. Cathey and Dowski [28] and Dowski and Cathey [41] propose a cubic phase plate design which yields a PSF that is broad-band spectrum and is relatively depth-invariant. Cossairt et al. [36] use a coded diffuser, which is a special type of phase plate, as shown in Figure 2.3 (c) for extended depth of field.

2.2.3 Sensor side coding

Sensor side coding (Figure 2.2 (c)) places additional optical elements on the sensor side of the lens. The element can be either placed in the space between the sensor and the lens, or placed on or close to the sensor, but in each case the functionality will be differ. According to the Gaussian lens law, optical devices after the lens are dual to devices in front of the lens, and therefore sensor side coding can provide similar functionalities as object side coding. One important advantage of using sensor side coding instead of object side coding is that it can be compactly built into a camera and hence is non-intrusive to the scene.

As in object side coding [49], lens arrays can also be used on the sensor side to capture light fields. The idea of the plenoptic camera has a long history that dates back to the early twentieth century [95] [76]. Since the 1990s, a variety of plenoptic cameras have been proposed and implemented in vision and graphics. Adelson and Bergen [3] proposed using a lenslet array in front of the sensor for light field acquisition. To achieve different amount of trade-offs between spatial and angular resolution, Lumsdaine and Georgiev [97] and Bishop et al. [18] proposed several different strategies of positioning lenslets and sensors.

Coding on the sensor plane provides pixel-wise modulations. Color filter arrays, such as the Bayer mode array, are widely used in these instances to encode color information in a monochromatic sensor [38] [14]. Other color filter patterns have also been proposed [96] [2], and various demosaicing algorithms have also been used to obtain a high quality color images [61] [65]. Nayar and Narasimhan [111] generalize the color filter array to assorted filter arrays in order to capture

extra multi-spectral and high dynamic range information.

2.2.4 Illumination coding

Illumination coding (Figure 2.2 (d)) alters captured images by using a spatially and/or temporally controllable camera flash. This approach enables image coding in ways that are not possible by only modifying the imaging optics. The basic function of the camera flash has remained the same since it first became commercially available in the 1930s. It is used to brightly illuminate scenes inside the camera FOV during the exposure time of the image detector. With significant advances made with respect to digital projectors, the flash now plays a more sophisticated role in capturing images. It enables the camera to project arbitrarily complex illumination patterns onto the scene, capture the corresponding images, and extract scene information that is not possible to obtain with a traditional flash.

Illumination coding has a long history in the field of computer vision. For example, virtually any structured light method (see [140] [141] for surveys) or a variant of photometric stereo [173] is based on the notion of illumination coding. Many other illumination coding techniques for depth estimation or 3D reconstruction have been proposed in recent years. Zhang and Nayar [180] and Gupta et al. [62] recover depth from defocused projections; and Kirmani et al. [78] measure the depths of points outside the camera's field of view by using echoes of pulsed illumination; Raskar et al. [130] use multiple flashes for depth edge measurement; Kinect depth sensor, a Microsoft gaming product released in 2010, combines an infrared projector with a monochrome CMOS sensor for 3D reconstruction [Microsoft].

Structured illumination techniques based on a phenomenon known as the Moiré effect have been used to overcome the resolution limits of microscopy [63] [64] and other imaging systems [24] [45] (see [148] for a survey of the Moiré technique). Structured illumination using diffuse optical tomography has been used for volume density estimation [74] [86].

2.2.5 Camera clusters or arrays

The capability of a single camera is virtually constrained by optical size, which physically determines the field of light to be captured. One way to transcend this limit is by using larger lenses. However, it is often too expensive and difficult to built large imaging systems of high quality. In

recent years, techniques have been proposed to use a number of low cost small cameras to capture more visual information. Camera clusters or arrays (Figure 2.2 (e)) provide a more flexible and economical way to transcend the limits of individual cameras by combining multiple cameras.

Camera arrays have been used for stereo vision over an extended history. Multi-view stereo helps to solve the ambiguity problem in stereo matching and hence increases the precision of depth estimation [66] [123] [11] [8] [52]. The high performance of camera arrays in HDR, FOV, synthetic aperture, and light field acquisition has been studied in [170] (shown in Figure 2.3 (d) left). A flexible array of cameras with divergent FOV is designed for scene collage [120] (see Figure 2.3 (d) right). Ding et al. [39] use a $3x3$ camera array to track distorted feature points beneath a fluid surface in order to dynamically recover fluid surfaces. In [151], an array of video cameras are used to stabilize video when the camera jitters.

2.2.6 Unconventional Imaging Systems

Unconventional coding (Figure 2.2 (f)) includes computational camera designs using unconventional architectures or non-optical devices that cannot fit well into the above five categories. Work has been done to simplify camera architectures by using computation instead of extending the functionalities of the camera. Stork and Robinson [155] and Robinson and Stork [136] discuss several mathematical and conceptual foundations for digital-optical joint optimization, and propose a singlet lens design and a triplet lens design with improved image quality after computation. Robinson and Stork [137] exploit the idea of digital and optical joint optimization for super-resolution.

It is also possible to change the overall architecture of cameras. For example, Zomet and Nayar [186] propose lensless cameras with one or multiple layers of controllable apertures for imaging. An XSlit camera by Zomet et al. [187] collects all rays that pass through two non-coplanar lines. Yu and McMillan [178] present a General Linear Camera (GLC) model that unifies many multiperspective cameras and reveal three new and previously unexplored multiperspective linear cameras by using the GLC model.

Among the six coding approaches, object side coding, pupil plane coding, and sensor side coding are modifications made to a traditional camera. Figure 2.4 gives an overview of the computational camera designs in these three categories. In the horizontal axis, we have object side coding, pupil plane coding, and sensor side coding. In the vertical axis, we have phase modulators (includ-

Devices		Object Side Coding	Pupil Plane Coding	Sensor Side Coding
Phase Modulators	Lens(es)	Lightfield: [52, 11]	Depth: [104]	Lightfield: [113,114,111,115,116,206]
	Prism(s) Plate(s)	Depth: [46,50,208,209] Color: [72]		
	Phaseplate		Depth: [98, 99] EDOF: [100,101,105,106,107,210]	
	Diffuser	Depth: [31] HDR: [77]	EDOF: [32,108]	
Intensity Modulators	Photomask	HDR: [33,70,215] Motion: [76]	Lightfield: [37,38] EDOF: [94,95] Depth: [34,80,81,82,83,84,89,90,217,218] Image: [34,35,36,85,86,87,88,211,212,213]	Lightfield: [35] HDR: [110,121,214]
	Color Filter	Color: [33]	Depth: [45]	Color: [43,116,117,119,120,121]
	Polarizer	Separation: [73,74,75,219,220]		
Others	Motion	EDOF: [132]	Depth: [221]	EDOF: [134] Image: [133,134,135] Motion: [76,112,136]
	Mirror(s)	Depth: [47,48,49,51,71,223,226] FOV: [53,54,55,56,57,58,59,60,64, 65,66,67,68,69,71,222,224,225]		HDR: [41,97,215] FOV: [41]

Color Scheme: Lightfield, Depth, Image, EDOF, HDR, Color, Separation, FOV, Motion

Figure 2.4: An overview of computational camera designs using object side coding, pupil plane coding, and sensor side coding. In the vertical direction are the optical devices that are often used in designing computational cameras. In each cell, we group the techniques according to the type of visual information to be captured, including light field, depth, image (i.e. spatial resolution), EDOF, HDR, Color, FOV, and motion (i.e. temporal resolution). Each group is differently colored. This table, although not exhaustive, provides an overview of existing computational camera designs and may inspire new ideas in this area.

ing lenses, lens arrays, prisms, prism arrays, plate, phaseplates (and diffusers), intensity modulators (including masks, color filters, and polarizers), and others (including mirrors and motions). Each cell groups the techniques according to the type of visual information being sought, including light field, depth, image (i.e. the spatial resolution), EDOF, HDR, color, FOV, and motion (i.e. the temporal resolution). This table, although not exhaustive, provides an overview of existing computational camera designs and may inspire new ideas in this exciting research area.

Chapter 3

PSFs for image deblurring

3.1 Introduction

Texture detail of a scene is often lost in a captured image due to defocus, lens aberration, or diffraction. As stated in the previous chapter, a blurry image can be often formulated as a convolution of the latent sharp image f_0 and a PSF k, plus noise η:

$$f = f_0 \otimes k + \eta, , \tag{3.1}$$

or in the Fourier domain,

$$F = F_0 \cdot K + \zeta, \tag{3.2}$$

where F_0, K and ζ are the discrete Fourier transforms of f_0, k, and η, respectively. The only way to recover scene details in blurry areas is by using deconvolution techniques, which is to estimate F_0 from F and K. The main problem with image deconvolution is that the higher frequencies of the signal are attenuated during image formation and consequently deconvolution amplifies image noise. For any given frequency in Fourier domain, the lower the power the blur kernel has, the greater the amplification of image noise.

Defocus is the most commonly seen image blur in photographs. For a traditional camera, an object will appear in-focus when it is on the focus plane and will appear blurry as it deviates from the focus plane. The shape of defocus PSF is determined by the aperture pattern and its scale is related to object depth. A traditional lens camera uses circular apertures and produces circular defocus PSFs that not only severely attenuate high frequencies but also have zero-crossings in frequency domain.

Figure 3.1: Defocus blurred image with a circular aperture and its deblurring result. (a) A focused image. (b) A defocused image captured using a circular (conventional) aperture. (c) The result of the deblurring. Ringing artifacts and the loss of image details can be easily observed (also see the zoomed inset images).

This has two adverse effects in the context of defocus deblurring - some frequencies simply cannot be recovered and image noise is greatly exaggerated. Figure 3.1 (b) shows a severely defocused image by a circular aperture and the result of deblurring. Ringing artifacts and the loss of image details can be easily observed.

Intuitively, a good defocus PSF should be broad-band in the frequency domain. Based on this intuition, people have proposed a variety of coded apertures for better defocus deblurring over the past 50 years (e.g. [169][101][165][122][26]). These works have evaluated and optimized aperture patterns based on intuitive criteria related to the shape of their power spectra. Although such intuitions have helped to find better aperture patterns, they are usually not quantitative and also do not explicitly account for the effects of image noise and image structure in the context of defocus deblurring. The exact connection between the defocus function and the final deblurring quality is absent in the literature.

In this thesis, we propose a criterion to evaluate the "goodness" of PSFs according to the expected quality of deblurring. In the criterion, the PSF spectrum is assessed together with the level of image noise and the expected spectrum of an image. Image prior such as the $1/f$ law [**?**][154][168] is also taken into account. This criterion is concise and in a close form, so that it can be easily used for camera optimization.

Since the shape of defocus PSF is determined by the pattern of lens aperture, we use the pro-

posed criterion to optimize aperture patterns for defocus deblurring. The optimized aperture patterns are shown to significantly outperform the circular aperture and other coded apertures in an extensive simulation. To experimentally verify the optimized patterns, we printed several aperture patterns as high resolution photomasks and inserted them into Canon EF $50mm$, $f/1.8$ lenses. These lenses were attached to a Canon EOS 20D camera and used to capture images of a wide variety of scenes.

3.2 Criterion for PSF quality: defocus deblurring

3.2.1 Formulating defocus deblurring

Given a defocused image F and known PSF K, the problem of defocus deblurring is to estimate the focused image F_0 by solving a maximum a posteriori (MAP) problem:

$$\hat{F_0} = \arg\max P(F_0|F, K) = \arg\max P(F|\hat{F_0}, K) \cdot P(\hat{F_0}). \tag{3.3}$$

By assuming a Gaussian model and then taking its logarithmic energy function, the above MAP problem can be solved as the minimization of

$$E(\hat{F_0}|F, K) = \|\hat{F_0} \cdot K - F\|^2 + H(\hat{F_0}). \tag{3.4}$$

The regularization term $H(\hat{F_0})$ can be formulated using a variety of image priors. To simplify our analysis, we constrain $H(\hat{F_0})$ to be $\|C \cdot \hat{F_0}\|^2$, where C is a matrix. Then, minimizing $E(\hat{F_0}|F, K)$ gives us the well-known Wiener deconvolution [10]:

$$\hat{F_0} = \frac{F \cdot \bar{K}}{|K|^2 + |C|^2}, \tag{3.5}$$

where $\bar{K}$ is the complex conjugate of K, $|K|^2 = K \cdot \bar{K}$, and $|C|^2 = C \cdot \bar{C}$. Furthermore, the optimal $|C|^2$ is known to be the matrix of noise-to-signal ratios (NSR), $|\sigma/F_0|^2$.

We generally do not have access to the exact NSR matrix since F_0 is unknown. The traditional approach is to replace $|C|^2$ with a single scalar parameter λ or a simplified matrix like $\lambda \cdot (|G_x|^2 + |G_y|^2)$, where G_x and G_y are the Fourier transforms of the spatial derivative filters in the x-axis and y-axis, respectively. These simplifications cause deconvolution to not be optimal. More importantly, the parameter λ needs to be tuned, which is difficult as it is inherently scene dependent.

3.2.2 Optimizing parameter C using an image prior

Since we would like our aperture pattern evaluation/optimization to be automatic, we seek a deconvolution method that is free of parameter selection. Given a blur pattern K and a defocused image F, the focused image can be estimated as $\hat{F_0}$ by using Equation (3.5). Since noise ζ is a random matrix, we evaluate the quality of recovery using the expectation of the L_2 distance between $\hat{F_0}$ and the ground truth F_0 with respect to ζ:

$$R(K, F_0, C) = \mathop{\mathbb{E}}_{\zeta}[\|\hat{F_0} - F_0\|^2] = \mathop{\mathbb{E}}_{\zeta}\left\|\frac{\zeta \cdot \bar{K} - F_0 \cdot |C|^2}{|K|^2 + |C|^2}\right\|^2, \tag{3.6}$$

where $\mathbb{E}$ denotes expectation. When ζ is assumed to be Gaussian white noise $N(0, \sigma^2)$, we have

$$R(K, F_0, C) = \left\|\frac{\sigma \cdot \bar{K}}{|K|^2 + |C|^2}\right\|^2 + \left\|\frac{F_0 \cdot |C|^2}{|K|^2 + |C|^2}\right\|^2. \tag{3.7}$$

Since F_0 is sampled from the space of all images and has a certain distribution, we look for a C that minimizes the expectation of R with respect to F_0:

$$R(K, C) = \mathop{\mathbb{E}}_{F_0}[R(K, F_0, C)] = \int_{F_0} R(K, F_0, C) d\mu(F_0), \tag{3.8}$$

where $\mu(F_0)$ is the measure of the sample F_0 in the image space. According to the $1/f$ law of natural images [104][154][168], we know that the expectation of $|F_0|^2$,

$$A(\xi) = \int_{F_0} |F_0(\xi)|^2 d\mu(F_0), \tag{3.9}$$

exists (ξ is the frequency). Therefore, we can obtain

$$R(K, C) = \left\|\frac{\sigma \cdot \bar{K}}{|K|^2 + |C|^2}\right\|^2 + \left\|\frac{A^{1/2} \cdot |C|^2}{|K|^2 + |C|^2}\right\|^2. \tag{3.10}$$

For a given K, minimizing $R(C|K)$ gives us

$$|C|^2 = \sigma^2 / A. \tag{3.11}$$

Therefore, by substituting $|C|^2 = \sigma^2/A$ into Equation 3.5, we have

$$\hat{F_0} = \frac{F \cdot \bar{K}}{|K|^2 + \sigma^2/A}. \tag{3.12}$$

In practice, A can be estimated by simply averaging the power spectra of several natural images. Since the noise level σ is determined by the camera model and its ISO (or gain) setting, this variant of Wiener deconvolution algorithm is parameter-free.

This is a variant of Wiener deconvolution augmented by using $1/f$ law of natural images. Although some people have already been using this algorithm in practice [133], we note that the significance of this algorithm is often overlooked and many people are still using the conventional Wiener deconvolution algorithm, in which C is set to be a scalar number.

3.2.3 Criterion of PSF evaluation for deblurring

A typical way to measure the quality of the recovered image $\hat{F}_0$ is to use its L_2 reconstruction error:

$$R = \|F_0 - \hat{F}_0\|^2. \tag{3.13}$$

From Equations 3.2 and 3.12, we can see that $\hat{F}_0$ is a function of F, K, and σ, and F depends on F_0, K, and ζ. Therefore, R is actually a function of F_0, K, and ζ, where ζ is the Fourier transform of Gaussian white noise $G(0, \sigma^2)$ and F_0 follows the $1/f$ law of natural images. Then, for a given PSF K, we can compute the expectation of R as:

$$R(K, \sigma) = E_{F_0,\zeta}(\|F_0 - \hat{F}_0\|^2) \tag{3.14}$$

$$R(K, \sigma) = \sum_{\xi} \frac{\sigma^2}{|K_\xi|^2 + \sigma^2/A_\xi}, \tag{3.15}$$

where ξ is the frequency. (See Appendix A for a detailed derivation.) $R(K, \sigma)$ predicts the deblurring quality if the aperture pattern K is used at a noise level σ and can be used as a criterion to evaluate aperture patterns.

For each frequency ξ, the reconstruction error $\frac{\sigma^2}{|K_\xi|^2+\sigma^2/A_\xi}$ is approximately proportional to $1/|K_\xi|^2$. This gives a clear explanation of why zero-crossings in the PSF spectrum will introduce large deblurring artifacts. In addition, $\|K_\xi\|^2$ falls off quickly as the frequency increases for most aperture patterns and σ^2/A_ξ increases relatively slowly. This explains why the high frequency part of images are more vulnerable to image noise than the low frequency part. While some other criteria such as $\Sigma\|K_\xi\|^2$ could be correct conceptually, our derived criterion is much more precise in predicting the deblurring quality.

The effect of image noise on the deblurring quality, which is almost completely overlooked by all previously introduced criteria, is now well described in Equation 3.15. We will show with more analyses that image noise plays a key role in defocus deblurring and should not be ignored in aperture evaluation and selection.

Table 3.1: Genetic Algorithm for Coded Aperture Optimization

1) Initial: g = 0; randomly generate S binary sequences of length L.

2) Repeat until g = G

a) *Selection*: For each sequence b, the corresponding blur function K is computed and then evaluated by using Equation 3.15. Only the best M out of S sequences are selected.

b) Repeat until the population (the number of sequences) increases from M to S.

— *Crossover:* Duplicate two randomly chosen sequences from the M sequences of Step 2.a, align them, and exchange each pair of corresponding bits with a probability of c_1, to obtain two new sequences.

— *Mutation:* for each newly generated sequence, flip each bit with a probability c_2.

c) g = g + 1.

3) Evaluate all the remaining sequences using Equation 3.15 and output the best one.

$*$ In our implementation, $L = 169$, $S = 4000$, $M = 400$, $c_1 = 0.2$, $c_2 = 0.05$ and $G = 80$.

3.3 PSF optimization for deblurring

3.3.1 Genetic algorithm for aperture optimization

We first use the derived criterion to solve for the optimal pattern for deblurring. However, even with our concise evaluation criterion in Equation (3.15), finding the optimal aperture pattern remains a challenging problem. While the aperture pattern is evaluated in the frequency domain, it must satisfy several physical constraints in spatial domain. For example, all its transmittance values must lie between 0 and 1; and the whole pattern should fit within the largest clear aperture of the camera. Deriving a closed-form optimal solution that satisfies all these constraints is difficult. We therefore resort to a numerical search approach.

For a binary pattern of resolution $N \times N$, the number of possible solutions is $2^{N \times N}$, making exhaustive search impractical even for small values of N. To solve this optimization problem, we develop a genetic algorithm [153]. Each aperture pattern k of size $N \times N$ is encoded as a binary sequential pattern b of length N^2. An aperture with significant discontinuities will produce strong

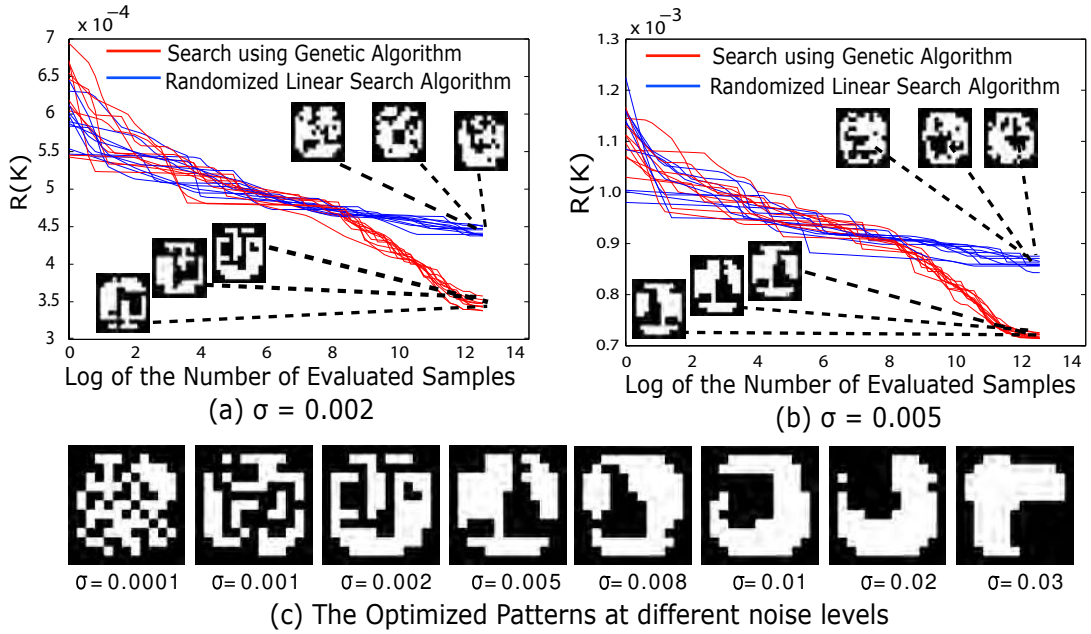


Figure 3.2: Optimizing Coded Aperture Patterns Using Genetic Algorithm. (a) Compare the convergence rates of optimization for $\sigma = 0.002$ between our proposed genetic algorithm (red) and a randomized linear search algorithm (blue). Each algorithm is repeated 10 times. (b) Compare the convergence rates for $\sigma = 0.005$. We see that our genetic algorithm converges quickly to a low value for aperture criterion metric. In addition, the results of different runs of the genetic algorithm are quite similar, indicating that they are all likely close to the optimum aperture. (c) shows the eight optimized patterns for noise levels from 0.0001 to 0.03. The patterns become more structured as the noise level increases.

diffraction effects. To this end, we limit the spatial resolution to be relatively low, i.e., N = 13.

To solve this optimization problem, we develop a genetic algorithm [153]. The process of this optimization algorithm is described in detail in Table 3.1. In our implementation, the population size in the first generation is set to $S = 4000$; at each generation, $M = 400$ sequences are selected for evolution; for crossover, each pair of corresponding bits in the parent sequences are switched with a probability of $c_1 = 0.2$; mutation defined as bit flipping, happens at each bit with a probability of $c_2 = 0.05$; and the evolution stops at the maximum number of generations, $G = 80$. The best sample (which gives the lowest value of the criterion in Equation 3.15, in the last generation corresponds to the optimal aperture pattern. For a 13×13 pattern, a total of $S \times G = 320,000$ samples are evaluated, which takes about 20 minutes on a 4GHz PC with our implementation.

Figure 3.2 compares the convergence rates for the genetic algorithm and a randomized linear search. We can see that for the genetic algorithm R drops quickly to a small number. To test if our optimization has converged to a "bad" local minimum, we repeated the optimization 10 times with different initial populations. While randomized linear searches always arrived at fairly different patterns, our genetic algorithm always converged to patterns with similar appearance. Although it is hard to prove, we believe this implies that our algorithm yields near-optimal solutions.

As stated earlier, the optimal aperture pattern varies with the level of image noise. We performed our optimization using eight levels of noise; σ = 0.0001, 0.001, 0.002, 0.005, 0.008, 0.01, 0.02, to 0.03. The resulting apertures are shown in the bottom row of Figure 3.2. It is interesting to note that the optimized aperture patterns become more structured with increase in noise.

3.3.2 Discussion

Optimized Patterns in Frequency Domain In Figure 3.3, we compare the Fourier spectrum of one of our optimized apertures ($\sigma = 0.001$) with that of the circular pattern, and Veeraraghavan et al.'s pattern in (a), and also compare it with other two optimized patterns ($\sigma = 0.005$ and 0.01) in (b). Though the figure only shows us a 1D slice of the 2D Fourier spectrum, it can give us a better intuition of how these apertures may work in out-of-focus deblurring. Figure 3.3 (a) shows that the circular pattern has many zero-crossings and greatly attenuates high frequencies, and thus may not be suitable patterns for deblurring; and (b) shows that the optimized pattern for small noise level tends to cover more high frequency parts, while the one optimized for large noise level has larger

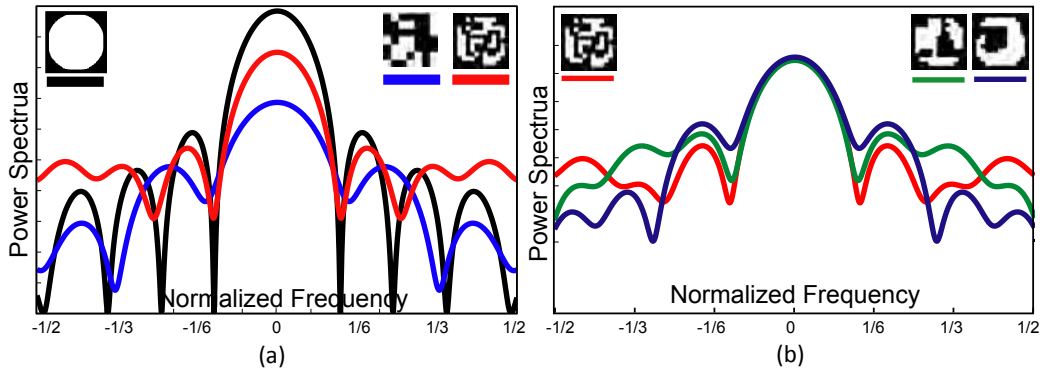


Figure 3.3: 1D slices of Fourier transforms of different patterns. (a) Circular pattern (black), Levin et. al.'s pattern (green), Veeraraghavan et. al.'s pattern (blue), and the optimized pattern for $\sigma = 0.001$ (red). (b) The optimized patterns for $\sigma = 0.001$ (red), $\sigma = 0.005$ (green), and $\sigma = 0.01$ (blue).

responses at low frequencies. Larger noise level means much less recoverable information in the high frequency part, hence the filter is optimized to put more emphasis in the low frequency part.

3.4 Experiments with real apertures

As shown in Figure 3.4(a), we printed our optimized aperture patterns as well as several other patterns as a single high resolution (1 micron) photomask sheet. To experiment with a specific aperture pattern, we cut it out of the photomask sheet and inserted it into a Canon EF $50mm\ f/1.8$ lens. In Figure 3.4(b), we show 4 lenses with different apertures (image pattern, Levin et al., Veeraraghavan et al, and one of our optimized patterns) inserted in them, and one unmodified (circular aperture) lens. Images of real scenes were captured by attaching these lenses to a Canon EOS 20D camera.

As previously mentioned, we choose the pattern that is optimized for $\sigma = 0.001$. This pattern exhibits high performance over a wide range of noise levels in the simulation. In addition, this Canon EF lens was found to produce some severe optical aberrations when operating with a fully open aperture ($f/1.8$). We therefore conducted our experiments with the lenses stopped down to $f/2.2$.

To calibrate the true PSF of each of the 5 apertures, the camera focus was set to $1.0m$; an array of point light sources was moved from $1.0m$ to $2.0m$ with $10cm$ increments; and an image

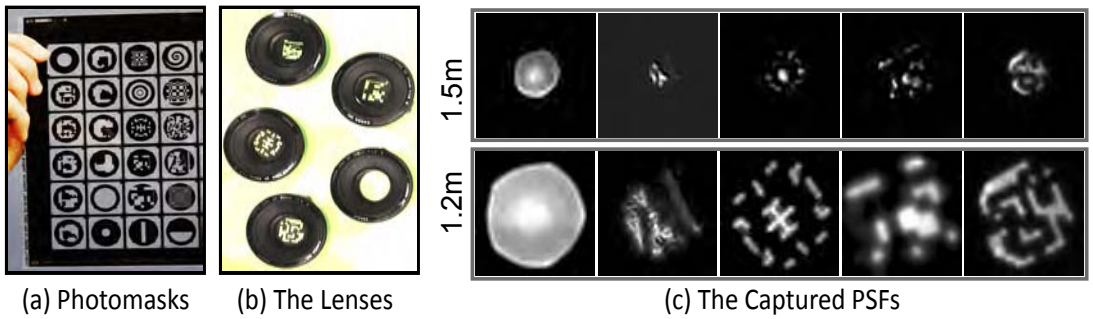


Figure 3.4: (a) Photomask sheet with many different aperture patterns. (b) One unmodified lens and four lenses with patterns inserted. (c) Top row shows calibrated PSFs for a depth of $120cm$ from the lens, and bottom row shows calibrated PSFs for a depth of $150cm$. These PSFs, from left to right, correspond to circular pattern, image pattern, Levin et al., Veeraraghavan et al., and one of our optimized patterns.

was captured for each position. Each defocused image of a point source was deconvolved using a calibrated focused image of the source. This gave us PSF estimates for each depth (source plane position) and several locations in the image. Since our lenses do not perfectly obey the thin lens model, the PSF was found to vary slightly over the image. In Figure 3.4(c-g), two calibrated PSFs (for depths of $120cm$ and $150cm$) are shown for each pattern. These PSFs correspond to the center of the image.

In our experiment, we placed a CZP resolution chart at the distance of $150cm$ from the lens, and capture images using the five different apertures. To be fair, the same exposure time was used for all the acquisitions. The five captured images and their corresponding deblurred results are shown in Figures 3.5. Notice that the captured images have different brightness levels as the apertures obstruct different amounts of light. The resulting brightness drop (compared to the circular aperture) for the image pattern, Levin et al., Veeraraghavan et al., and our optimized pattern are 52%, 48%, 35%, and 57%, respectively.

Note that our optimized pattern gives the sharpest deblurred image with least artifacts and image noise. We have conducted a quantitative analysis to compare the performances of the five apertures. We carefully aligned all the deblurred images to the focused image with sub-pixel accuracy, and computed their residual errors. The residual errors are then analyzed in frequency domain. In

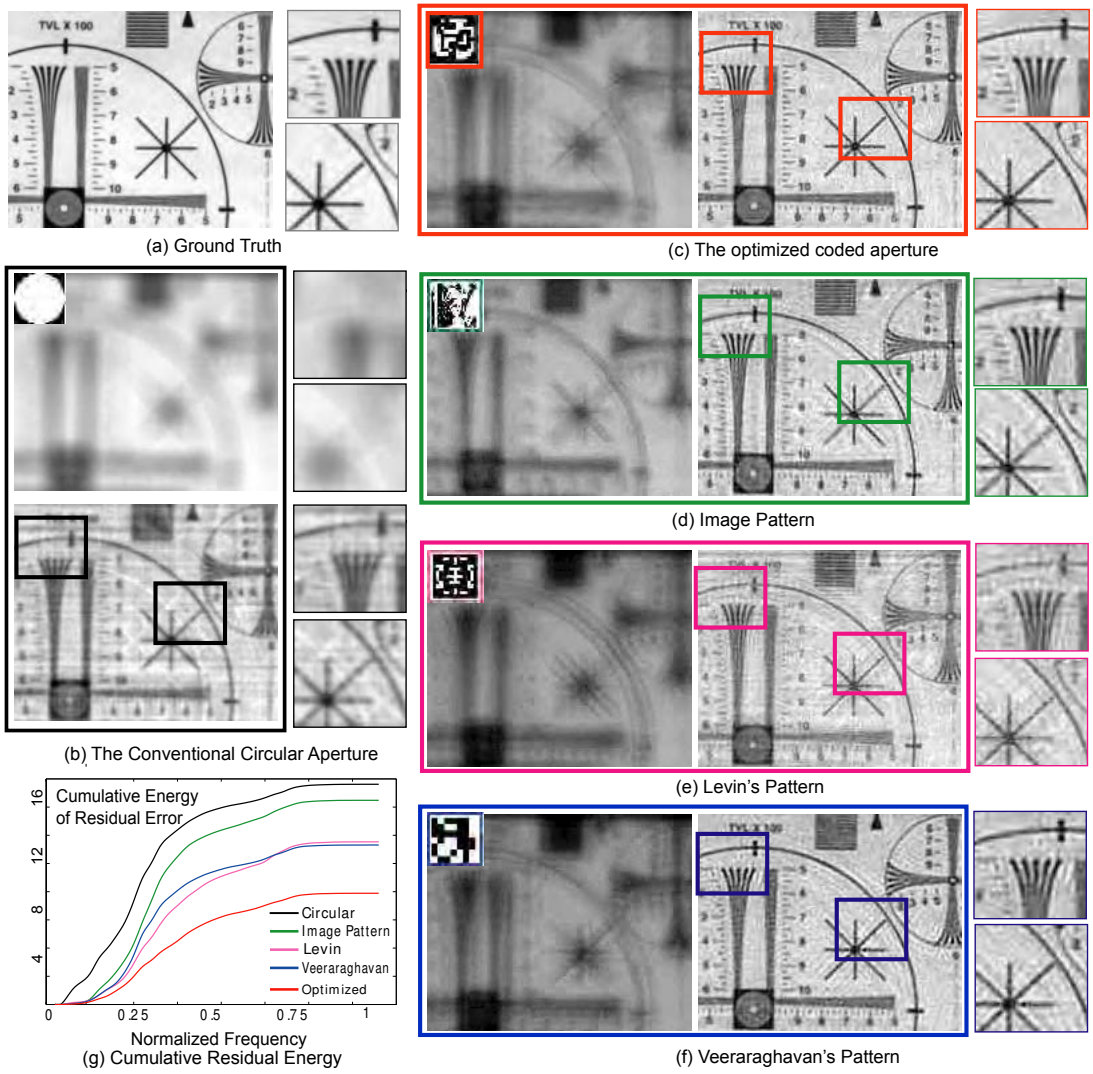


Figure 3.5: Comparison between deblurring of a CZP resolution chart using different apertures. (a) A focused image. (b) The captured and deblurred images using a conventional circular aperture. (c-f) The left shows captured (defocused) images and the right shows the deblurred images, for four different aperture patterns, including one of our optimized patterns, an image pattern, Levin's pattern, and Veeraraghavan's pattern. Both the captured images were taken under the same focus setting and the same exposure time. The deblurred image in (c) is clearly of higher quality than the ones in (b, d-f). (g) For each aperture, the cumulative energy of the residual error between the ground truth and deblurred images is plotted as a function of frequency.

Figure 3.5(d), we plot the cumulative energy of the residual error from low to high frequency. The image pattern, Levin et al., and especially Veeraraghavan et al., show large improvements over the circular aperture. Our optimized aperture is seen to produce the lowest residual error with about 30% improvement over Veeraraghavan et al. (which performs the best among the rest).

3.4.1 Deblurring Results for Complex Scenes

We have used the lens with our optimized aperture pattern to capture several complex real scenes with severely defocused regions (see Figure 3.6). We then applied deblurring to the defocused regions. Deblurring of a region requires prior knowledge of its depth. In all our examples, the user interactively selected the depth that produced the most appealing deblurring results. This is made possible by the fact that the deblurring algorithm described in Section 3.2.1 is very fast and requires no parameter selection. For a 1024×768 image, our Matlab implementation of the algorithm takes only 30 seconds to test 20 depths. In contrast, other deblurring algorithms that use sparse image priors can take 30 mins for a single depth, not to mention the time needed to adjust parameters.

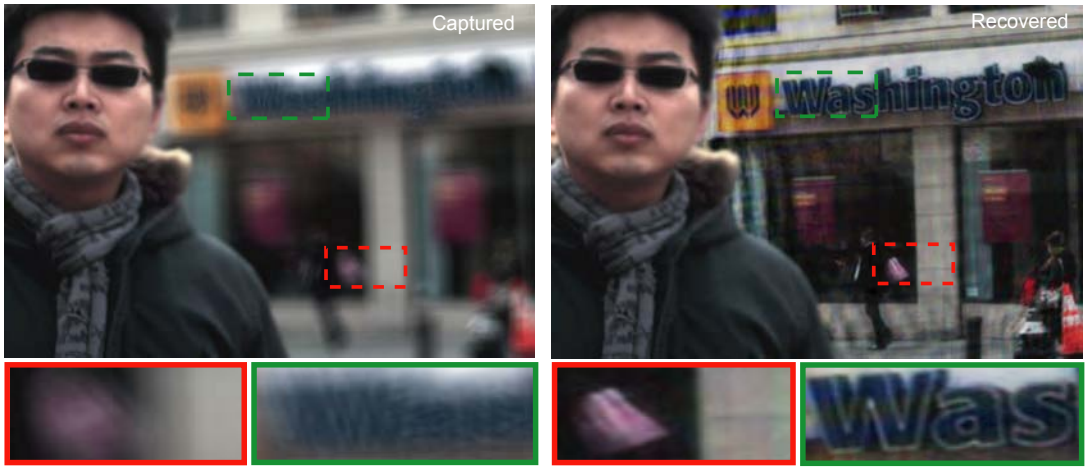
Figures 3.6(a) and (b) show captured images (left) for which the camera was focused on the foreground object, making the background (poster in (a), and building and pedestrians in (b)) severely defocused. To deblur the background, we first segmented out the foreground region, filled the resulting hole using inpainting, and then applied deblurring using 40 different depths. The best deblurred result is chosen and merged with the foreground. Figure 3.6(c) shows a traffic scene where all the objects are out of focus. In this case, the final result was obtained using four depth layers. Although some ringing artifacts can be seen in our deblurred images, a remarkable amount of details are recovered in all cases. Please note the defocus in our experiments is much more severe than that in most other related works. For example, the recovered telephone number and taxi number in Figure 3.6(c) are virtually invisible in the captured image.

3.4.2 Coded aperture implementation with LCoS

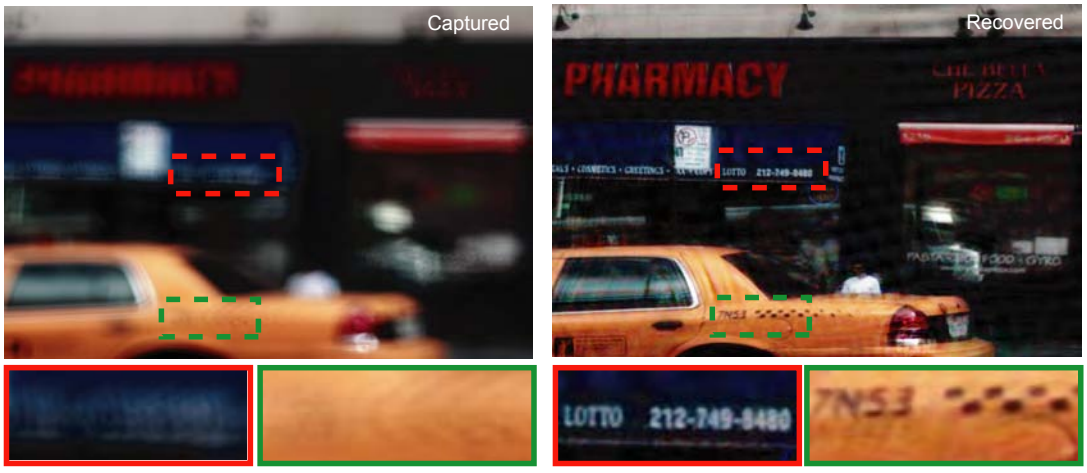
One important observation of our analysis is that the optimal aperture varies with the level of image noise. It will be ideal to have a camera with a programmable aperture, which will allow us change aperture pattern dynamically according to the variation of scene and application.

One programmable aperture implementation that is often used in the literature is to use a liquid

(a) Indoor Scene

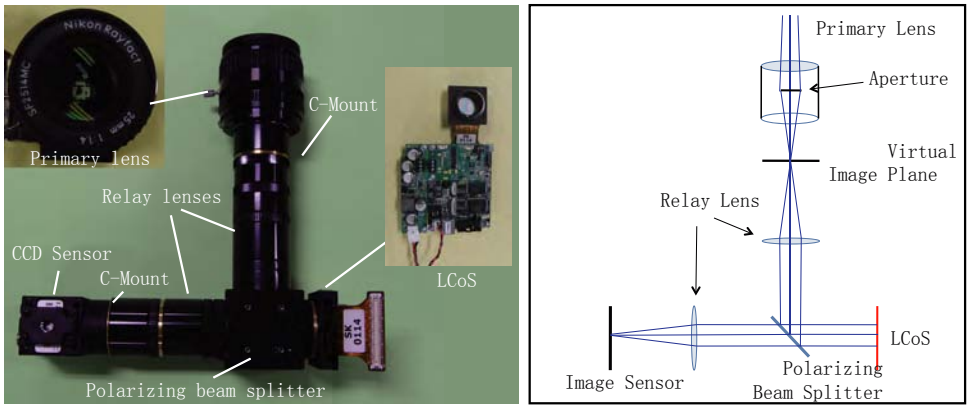


(b) Pedestrian Scene



(c) Traffic Scene

Figure 3.6: Deblurring results for three complex scenes. Left: Captured images with close-ups of several regions which are severely defocused; Right: Deblurring results with close-ups of the corresponding regions.



(a) A prototype programmable aperture camera (b) The optical diagram

Figure 3.7: Programmable aperture camera using an LCoS device. (a) A prototype LCoS programmable aperture camera. In the left-top corner is the Nikon F/1.4 25mm C-mount lens that is used in our experiments. You can see the aperture pattern inside the lens. On the right is an LCoS device. (b) The optical diagram of the proposed LCoS programmable aperture camera.

crystal display as aperture [186] [93]. However, this LCD implementation has many drawbacks. The liquid crystal occludes more than 75% of light; the electronic element in each pixel of LCD leads to complicated and strong defocus and diffraction artifacts; the liquid crystal display cannot provide high brightness contrast. Furthermore, it is often difficult to open the lens and insert an LCD into the aperture plane. These drawbacks are so strong that it may completely eliminate the benefits of aperture coding.

We therefore have worked with colleagues in Osaka University in building a programmable aperture camera using a Liquid Crystal on Silicon (LCoS) device [**?**]. LCoS is a reflective liquid crystal device that has a much higher fill factor (92%) than the transmissive ones, such as LCD. Compared with LCD, an LCoS device usually suffers much less from light loss and diffraction. Figure 3.7 shows the structure of our proposed programmable aperture camera. The use of LCoS device in our prototype camera enables us to dynamically change aperture patterns as needed at a high resolution (1280×1024 pixels), a video frame rate (25 fps), and a high brightness contrast (221:1). By using the relay optics, we can mount any C-Mount or Nikkon F-Mount lens to our programmable aperture camera.

In our experiment, we select the pattern shown in Figure 3.8 from our optimized patterns (Figure

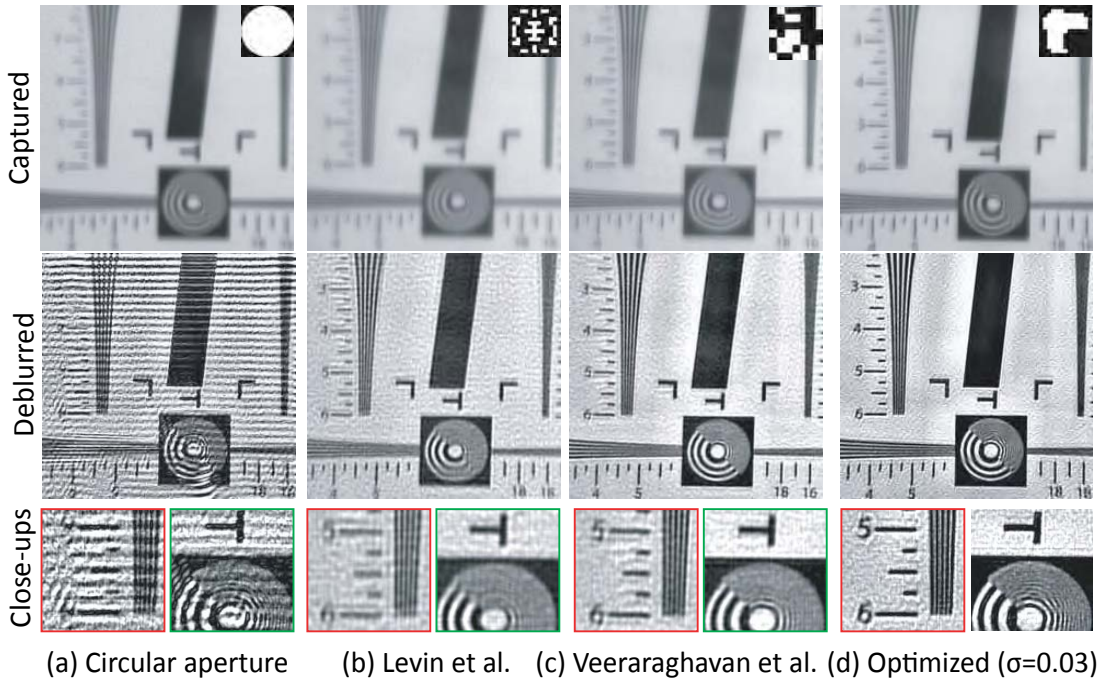


Figure 3.8: Defocus deblurring with coded apertures by using a programmable aperture camera. We select the pattern shown in Column (d) from our optimized patterns (Figure 3.2 (c)) according to the image noise level. We compare the selected pattern with the traditional circular aperture (a), the pattern designed by Levin et al. [88] (b), and the pattern designed by Veeraraghavan et al. [166] (c). The top row are the captured defocused images with the aperture pattern shown in the right-top corner; the second row are the deblurred images; and in the third raw we show close-ups of the deblurring results.

3.2 (c)) according to the image noise level. We compare the selected pattern with the traditional circular aperture (a), the pattern designed by Levin et al. [88] (b), and the pattern designed by Veeraraghavan et al. [166] (c). We capture a set of defocused images of an IEEE resolution chart and do image deblurring. The top row are the captured defocused images with the aperture pattern shown in the right-top corner; the second row are the deblurred images; and in the third raw we show close-ups of the deblurring results. We can see that the deblurring result in Column (d) is the best, which is consistent with the prediction.

It should be noted that the prototype camera is built to verify the concept of programmable

aperture camera and is still far from being ideal. For example, we used two doublet lenses to relay lights in this prototype camera and this causes significant image distortion and field curvature. Many of the optical imperfections can be solved by using better optical designs.

3.5 Summary

In this chapter, we answer the question of "What are good PSFs for defocus deblurring?", by presenting a comprehensive framework for PSF evaluation. Our derived evaluation criterion predicts the expected reconstruction error of the deblurred images, accounting for the effects of image noise as well as the statistics of natural images. We define the deblurring quality as the L_2 reconstruction error and constrain our discussion to linear deconvolution algorithms in order to make many analytical derivations possible. We have used the $1/f$ law as a prior for natural images. This prior, although not as strong as some other sparsity priors, is quite robust for a variety of natural images.

Chapter 4

PSFs for depth from defocus

4.1 Introduction

While defocus causes a loss in image details, it also encodes depth information of the scene. Depth from defocus (DFD) is a typical approach to recovering 3D scene geometry from defocus that has received renewed attention in recent years. For a given camera setting, scene points at greater distances away from this focal plane will appear increasingly blurred due to defocus, i.e. the scale of PSF increases with the distance from the focal plane, as illustrated Figure 1.2. By capturing two images at camera settings with different defocus characteristics, one can infer the depth of each point in the scene from their relative defocus. Relative to other image-based shape reconstruction approaches such as multi-view stereo, structure from motion, range sensing and structured lighting, depth from defocus is more robust to occlusion and correspondence problems [144].

Since defocus information was first used for depth estimation in the early 1980's [124][156], various techniques for DFD have been proposed based on changes in camera settings. Most commonly, DFD is computed from two images acquired from a fixed viewpoint with different aperture sizes (e.g., [116] [129] [167] [43]). Since the lens and sensor are fixed, the focal plane remains the same for both images. The image with a larger aperture will exhibit greater degrees of defocus with respect to given distances from the focal plane, and this difference in defocus is exploited to estimate depth.

Though most DFD methods employ conventional lenses whose apertures are circular, other aperture structures can significantly enhance the estimation of relative defocus and hence improve

depth estimation. In this thesis, we propose a comprehensive framework of evaluating aperture pairs for DFD, and use it to solve for an optimized pair of apertures. First, we formulate DFD as finding a depth d that minimizes a cost function $E(d)$, whose form depends upon the aperture patterns of the pair. Based on this formulation, we then solve for the aperture pair that yields a function $E(d)$ with a more clearly defined minimum at the ground truth depth d^*, which leads to higher precision and stability of depth estimation. Note that there exist various other factors that influence the depth estimation function $E(d)$, including scene content, camera focus settings, and even image noise level. Our proposed evaluation criterion takes all these factors into account to find an aperture pair that provides improved DFD performance.

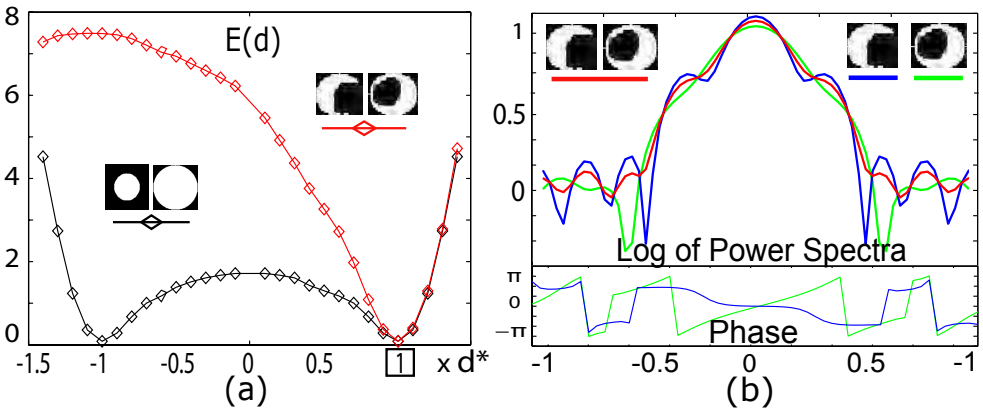


Figure 4.1: Depth estimation curves and pattern spectra. (a) Curves of $E(d)$ for the optimized coded aperture pair (red) and the conventional large/small circular aperture pair (black). The sign of the x-axis indicates if a scene point is farther or closer than the focus plane. (b) Top: Log of combined power spectra of the optimized coded aperture pair (red), as well as the power spectra of each single coded aperture (green and blue). Bottom: Phases of the Fourier spectra of the two coded apertures.

Solving for an optimized aperture pattern is a challenging problem as stated in the previous chapter. To make this problem more tractable, existing methods [182][166][88] have limited the pattern resolution to 13×13 or lower. However, solutions at lower resolutions are less optimal due to limited flexibility. To address the aperture resolution issue, we propose a novel recursive pattern optimization strategy that incorporates a genetic algorithm [182] with gradient descent search. This algorithm yields optimized solutions with resolutions of 33×33 or higher within a reasonable computation time. Although higher resolutions usually mean greater diffraction effects, in this

particular case, we find that a high-resolution pattern of 33×33 suffers less from diffractions than other lower resolution patterns do.

Figure 4.1(a) displays profiles of the depth estimation function $E(d)$ for the optimized pair and for a pair of conventional circular apertures. The optimized pair exhibits a profile with a more pronounced minimum, which leads to depth estimation that has lower sensitivity to image noise and greater robustness to scenes with subtle texture. In addition, our optimized apertures are found to have complementary power spectra in the frequency domain, with zero-crossings located at different frequencies for each of the two apertures, as shown in Figure 4.1(b). Owning to this property, the two apertures thus jointly provide broadband coverage of the frequency domain. This enables us to also compute a high quality all-focused image from the two captured defocused images.

We demonstrate via simulations and experiments the benefits of using an optimized aperture pair over other aperture pairs, including circular ones. Our aperture pair is able to not only produce depth maps of significantly greater accuracy and robustness, but also produces high-quality all-focused images (see Figure 4.2 for an example.)

4.2 Criterion for PSF quality: depth from defocus

4.2.1 Formulation of depth from defocus

As shown in Equation 3.1, for a simple fronto-planar object, its out-of-focus image can be expressed as the convolution of in-focus image and PSF, plus noise. A single defocused image is generally insufficient for inferring scene depth without additional information. For example, one cannot distinguish between a defocused image of sharp texture and a focused image of smoothly varying texture. To resolve this ambiguity, two (or more) images of a scene are conventionally used, with different defocus characteristics or PSFs for each image:

$$F_i = F_0 \cdot K_i^{d^*} + \zeta_i, \tag{4.1}$$

where $K_i^{d^*}$ denotes the Fourier transform of the i^{th} PSF with the actual blur size d^*. Our objective is to find the size $\hat{d}$ and deblurred image $\hat{F}_0$ that minimize the following energy function:

$$E(\hat{d}, \hat{F}_0 | F_1, F_2) = \sum_{i=1,2} \|\hat{F}_0 \cdot K_i^{\hat{d}} - F_i\|^2 + \|C \cdot \hat{F}_0\|^2, \tag{4.2}$$

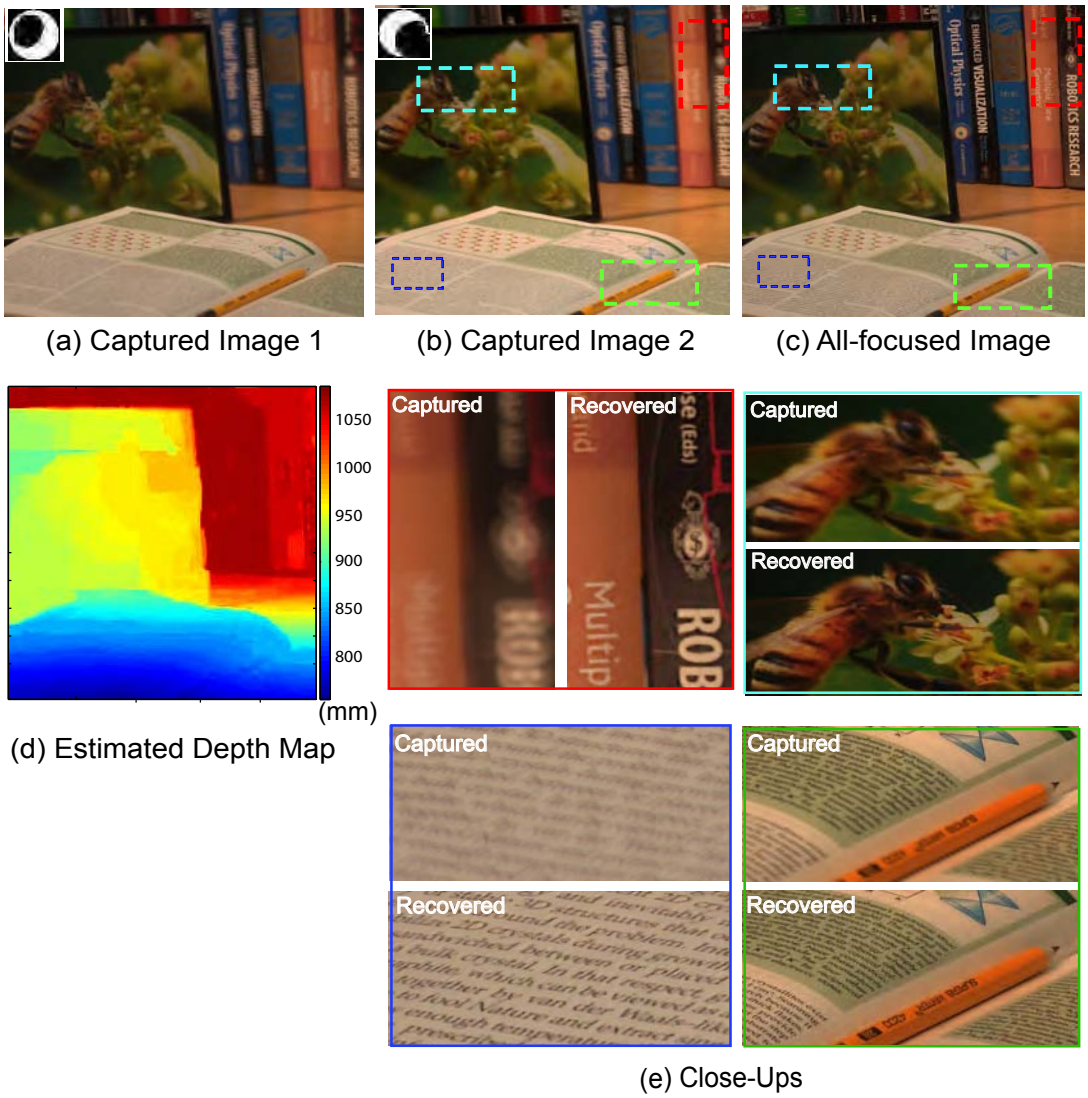


Figure 4.2: Depth from defocus and out-of-focus deblurring using coded aperture pairs. (a-b) Two captured images using the optimized coded aperture pair. The corresponding aperture pattern is shown at the top-left corner of each image. (c) The recovered all-focused image. (d) The estimated depth map. (e) Close-ups of four regions in the first captured image and the corresponding regions in the recovered image. Note that the bee and flower within the picture frame (light blue box) are out of focus in the actual scene and this blur is preserved in the computed all-focused image. For all the other regions (red, blue, and green boxes) the blur due to image defocus is removed.

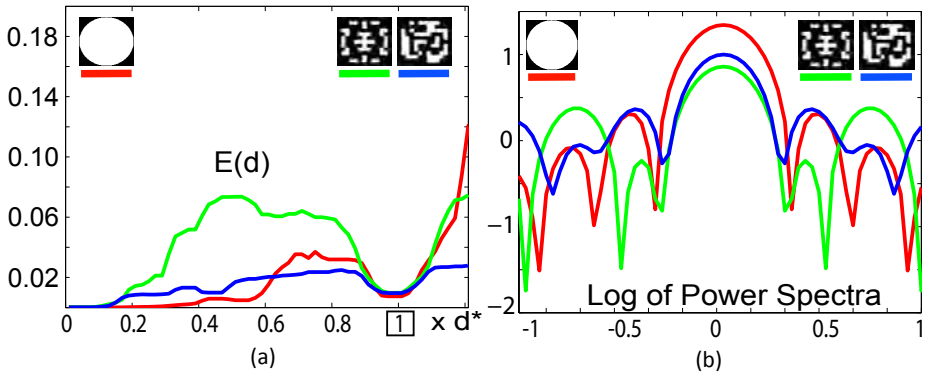


Figure 4.3: Performance trade-offs with single apertures. (a) DFD energy function profiles of three patterns: circular aperture (red), coded aperture of [88] (green), and coded aperture of [182] (blue). (b) Log of power spectra of these three aperture patterns. The method of [88] provides the best DFD, because of its distinguishable zero-crossings and its clearly defined minimum in the DFD energy function. On the other hand, the aperture of [182] is best for defocus deblurring because of its broadband power spectrum, but is least effective for DFD due to its less pronounced energy minimum, which makes it more sensitive to noise and weak scene textures.

in which the first term represents error in the solution with respect to the input images, and the second regularization term penalizes deviation of the deblurred image from an image prior. As shown in Chapter 1, C is the matrix of noise-to-signal ratios $\sigma/A^{\frac{1}{2}}$, where A is defined over the power distribution of natural images according to the $1/f$ law [168]: $A(\xi) = \int_{F_0} |F_0(\xi)|^2 \mu(F_0)$. Here, ξ is the frequency and $\mu(F_0)$ is the possibility measure of the sample F_0 in the image space. For a given $\hat{d}$, solving $\partial E/\partial \hat{F_0} = 0$ yields

$$\hat{F_0} = \frac{F_1 \cdot \bar{K}_1^{\hat{d}} + F_2 \cdot \bar{K}_2^{\hat{d}}}{|K_1^{\hat{d}}|^2 + |K_2^{\hat{d}}|^2 + |C|^2}, \tag{4.3}$$

where $\bar{K}$ is the complex conjugate of K and $|X|^2 = X \cdot \bar{X}$. Equation (4.3) can be regarded as a generalized Wiener deconvolution which takes two input defocused images, each with a different PSF, and outputs one deblurred image. This algorithm yields much better deblurring results than only deconvolving one input image [156] [88]. Note that a similar deconvolution algorithm was derived using a simple Tikhonov regularization in [126].

Substituting Equation (4.3) into Equation (4.2), we obtain the objective function $E(\hat{d}|K_1, K_2, \sigma, F_1, F_2)$.

Then, depth is estimated as the $\hat{d}$ that minimizes $E(\hat{d})$:

$$\hat{d} = \arg\min_{d} E(d|K_1, K_2, \sigma, F_1, F_2). \tag{4.4}$$

4.2.2 Selection criterion

Based on the above formulation of DFD, we seek a criterion for selecting an aperture pair that yields precise and reliable depth estimates. For this, we first derive $E(d|K_1^{d^*}, K_2^{d^*}, \sigma, F_0)$ by substituting Equations (4.1) and (4.3) into Equation (4.2). Note that the estimate d is related to the unknown F_0 and the noise level σ. We can integrate out F_0 by using the $1/f$ law of natural images as done in the previous chapter:

$$E(d|K_1^{d^*}, K_2^{d^*}, \sigma) = \int_{F_0} E(d|K_1^{d^*}, K_2^{d^*}, \sigma, F_0)\mu(F_0).$$

This equation can be rearranged and simplified to get

$$\begin{aligned} E(d|K_1^{d^*}, K_2^{d^*}, \sigma) = &\textstyle\sum_\xi \frac{A\cdot|K_1^d\cdot K_2^{d^*} - K_2^d\cdot K_1^{d^*}|^2}{\sum_i |K_i^d|^2 + C} \\ &+ \textstyle\sum_\xi \frac{\sigma^2\cdot(\sum_i |K_i^{d^*}|^2 + C)}{\sum_i |K_i^d|^2 + C} + n\cdot\sigma^2, \end{aligned} \tag{4.5}$$

which is the energy corresponding to a hypothesized depth estimate given the aperture pair, focal plane and noise level (see Appendix B for detailed derivations.)

The first term of Equation (4.5) measures inconsistency between the two defocused images when the estimated depth d deviates from the ground truth d^*. This term will be zero if $K_1 = K_2$ or $d = d^*$. The second term relates to exaggeration of image noise due to inaccurate depth estimation, and is minimized when $d = d^*$.

Depth can be estimated with greater precision and reliability if $E(d|K_1^{d^*}, K_2^{d^*}, \sigma)$ increases significantly when d deviates from the ground truth depth d^*. To ensure this, we evaluate the aperture pair (K_1, K_2) at depth d^* and noise level σ using

$$\begin{aligned} &R(K_1, K_2|d^*, \sigma) \\ &= \min_{d\in\mathcal{D}/d^*} E(d|K_1^{d^*}, K_2^{d^*}, \sigma) - E(d^*|K_1^{d^*}, K_2^{d^*}, \sigma) \\ &= \min_{d\in\mathcal{D}/d^*} \sum_\xi A\frac{|K_1^d K_2^{d^*} - K_2^d K_1^{d^*}|^2}{\sum_i |K_i^d|^2 + C} + \sigma^2\frac{\sum_i |K_i^{d^*}|^2 - \sum_i |K_i^d|^2}{\sum_i |K_i^d|^2 + C}, \end{aligned} \tag{4.6}$$

where $\mathcal{D}$=$\{c_1 d^*, c_2 d^*, \ldots, c_l d^*\}$ is a set of depth samples. In our implementation, $\{c_i\}$ is set to $\{0.1, 0.15, \ldots, 1.5\}$.

According to the derivations, this criterion for evaluating aperture pairs is dependent on depth d^* and noise level σ. However, this dependence is actually weak. Empirically, we have found Equation (4.5) is dominated by the first term, and C to be negligible in comparison to the other factors. As a result, Equation (4.5) is relatively insensitive to the noise level, such that the dependence on σ can be disregarded in the aperture pair evaluation (σ is taken to be 0.005 throughout this chapter).

We then standardize Equation 4.6 and get

$$R \approx \min_{d \in \mathcal{D}/d^*} \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1^d K_2^{d^*} - K_2^d K_1^{d^*}|^2}{|K_1^d|^2 + |K_2^d|^2 + C^2} \right]^{1/2}, \tag{4.7}$$

where n is the pixel number of the PSF. Let

$$M(K_1, K_2, d, d^*) = \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1^d K_2^{d^*} - K_2^d K_1^{d^*}|^2}{|K_1^d|^2 + |K_2^d|^2 + C^2} \right]^{1/2}, \tag{4.8}$$

then we have

$$R = \min_{d \in \mathcal{D}/d^*} M(K_1, K_2, d, d^*). \tag{4.9}$$

A larger R value indicates the energy function for DFD is steeper and therefore the estimation will be more robust to weak texture and image noise.

4.2.2.1 Analysis

When the ratio $c = d/d^*$ approaches to 1, we have

$$\begin{aligned} & M(K_1, K_2, d, d^*) \\ =& \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{(|c-1|d^*)^2 |K_1'^{d^*} K_2^{d^*} - K_2'^{d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2} \right]^{1/2} \\ =& |c-1| d^* \cdot \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1'^{d^*} K_2^{d^*} - K_2'^{d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2} \right]^{1/2}, \end{aligned} \tag{4.10}$$

where $K_i'^{d^*}$ is the derivative of $K_i^{d^*}$ with respect to the blur size. See Appendix C for the detailed derivation. It indicates that the M curve is linear to c when $|c| \to 1$. For a specific d^* and frequency ξ, the slope is determined by $\frac{|K_1'^{d^*} K_2^{d^*} - K_2'^{d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2}$. Figure 4.4 (a) shows M curves of a circular aperture pair at three different depths. We can see that the M curves are linear when $d \to d^*$.

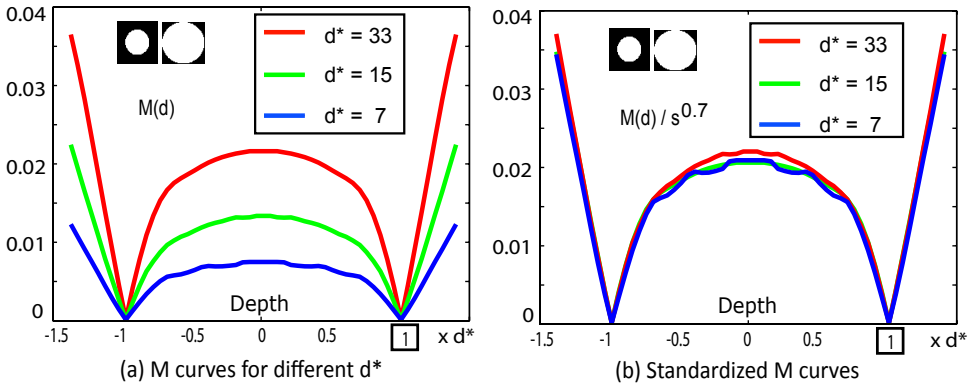


Figure 4.4: M curves. (a) Three M curves of a circular aperture pair at $d^* = 33, 15$, and 7 pixels, plotted as red, green, and blue lines, respectively. When $d \to d*$, the M curves are linear to d. (b) Three standardized M curves. Note the normalization factor $s^{0.7}$ does not rely on specific aperture patterns (Equation 4.11). The three standardized M curves are quite consisitent. It indicates the proposed evaluation criterion works equally well for different scene depths. Once an aperture pair is optimized for a specific blur size d^* (i.e. a specific object depth), it will also be optimal for other depths.

For optimal DFD performance with an aperture pair, intuitively, the pair must maximize the relative defocus between the two images. Equation 4.10 reveals that defocus depends on differences in amplitude and phase in the spectra of the two apertures. DFD is most accurate when the two Fourier spectra are complementary in both magnitude and phase, such that their phases are orthogonal and a zero-crossing for one aperture corresponds to a large response at the same frequency for the other aperture. For example, if $K_1 = 0$ at a specific frequency ξ, the slope

$$\frac{|K_1'^{d^*} K_2^{d^*} - K_2'^{d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2} = |K_1'^{d^*}|^2 \cdot \frac{|K_2^{d^*}|^2}{|K_2^{d^*}|^2 + C^2}.$$

Then, a larger derivative of K_1 and a larger $|K_2|$ are preferred at this frequency to maximize the slope. As a result, although our main objective is to compute optimal apertures for DFD, the complementary power spectra yielded by our approach also enables the capture of a broad range of scene frequencies and hence is effective for defocus deblurring.

Differences in d^* correspond to variations in the size of ground truth PSF, which is in turn determined by the depth. To assess how the depth variation affects the aperture pair evaluation, consider two PSF scales d_1^* and d_2^* with a ratio $s = d_2^*/d_1^*$. By assumping that the ratio $c = d/d^*$ approaches to 1 as we derive Equation 4.10, we are able to get

$$M(K_1, K_2, c \cdot d_2^*, d_2^*) \approx M(K_1, K_2, c \cdot d_1^*, d_1^*) \cdot s^{\alpha/2}, \tag{4.11}$$

where α is a constant number that is related to the power order in the $1/f$ law [164]. See Appendix D for the detailed derivation. Note the factor $s^{\alpha/2}$ is dependent of the choice of aperture patterns. Figure 4.4 (b) shows three standardized M curves of the circular aperture pair by factors $s^{\alpha/2}$. In our implementation, α is found to be 1.4. We can see the three M curves are quite consitent after the standardization. This indicates our evaluation criterion works equally well for all scene depths. This property ensures that once an aperture pair is optimized for a specific blur size d^* (i.e. a specific object depth), it will also be optimal for other depths.

In these analysis, the proposed criterion (Equation 4.9) is simplified by assuming $d/d^* \to 1$. While this helps us better understand the criterion in an intuitive way, it is not accurate when d is significantly different from d^*. For example, as shown in Figure 4.1, M is not longer linear to c when $|c|$ deviates far away from 1. Because of this, we will still use Equation 4.9 as the criterion for aperture pair evaluation.

4.2.3 Circular aperture pair

We first use our derived evaluation criterion to determine the optimal radius ratio of circular aperture pairs for DFD. In Figure 4.5 (a), we show curves of the M energy function from Equation (4.8) for four different ratios. These plots highlight the well-known ambiguity with circular aperture pairs of whether a scene point lies in front of or behind the focal plane. This problem exists for any point-symmetric apertures (e.g. the one optimized in [88]). Figure 4.5 (b) shows a plot of our evaluation measure R with respect to the radius ratio. R is maximized at the ratio 1.5, which indicates 1.5 is the optimal radius ratio for DFD.

4.3 PSF optimization for DFD

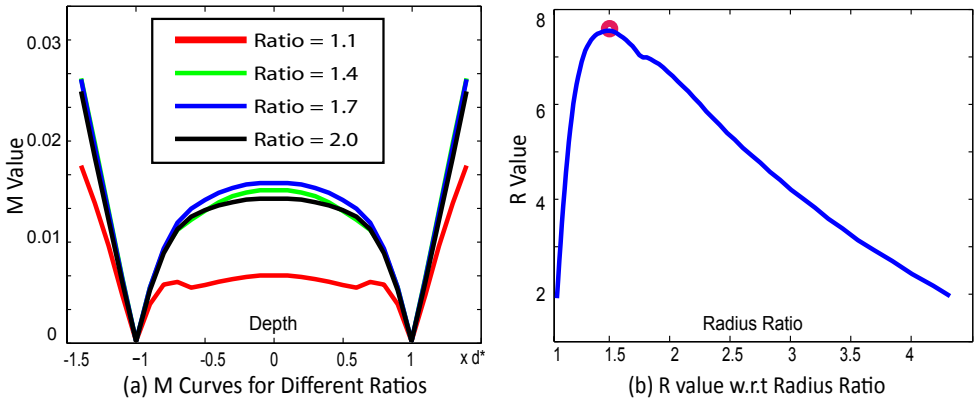


Figure 4.5: Using M and R to determine optimal radius ratio for DFD in the case of the conventional circular aperture. (a) M curves of the circular aperture pairs with four different radius ratios. (b) R values of circular aperture pairs with respect to radius ratio. R value is maximized at a radius ratio of 1.5.

A related analysis specifically for Gaussian aperture patterns has been previously performed in [129] and an optimal ratio of 1.73 was derived based on information theory. For Gaussian PSFs, our numerical optimization yields a similar ratio of 1.70. This shows the consistency between the theoretical approach and our numerical approach. While this theoretical approach requires Gaussian PSFs, our method can be applied to optimize arbitrary patterns.

4.3.1 Coded aperture pair

We then use the evaluation criterion to solve for optimal coded aperture patterns. Pattern optimization is known to be a challenging problem as stated in the previous chapter. For $N \times N$ binary patterns, the number of possible solutions is huge, $2^{N \times N}$. If we use gray-level patterns, the space will be even larger. Our problem is made harder since we are attempting to solve for a pair of apertures rather than a single aperture. To solve this problem, we propose a two-step optimization strategy.

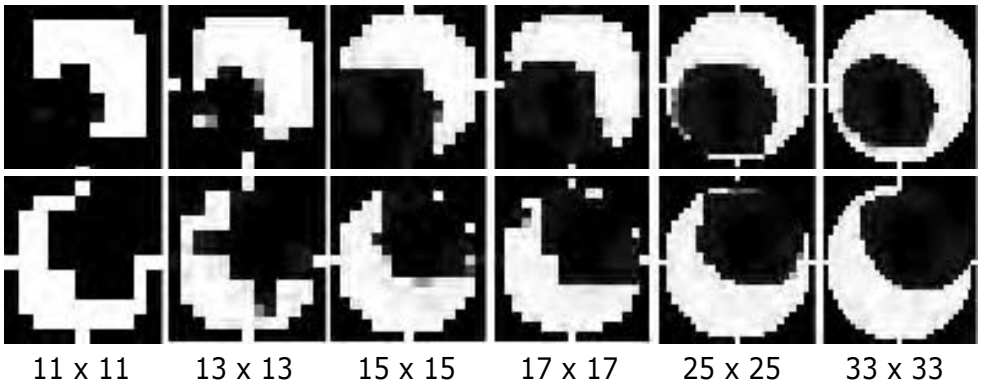


Figure 4.6: Increasing the resolution of an optimized aperture pair by up-sampling and gradient search.

In the first step, we employ the genetic algorithm proposed in the previous chapter to find the optimized binary aperture at a low resolution of 11×11 according to Equation (4.9). The optimized aperture pair at 11×11 is shown in the first column of Figure 4.6. Despite the high efficiency of this genetic algorithm, we found it to have difficulties in converging at higher resolutions.

As discussed in Section 4.2.2.1, the optimality of an aperture pair is invariant to scale. Therefore, scaling up the optimized pattern pair yields an approximation to the optimal pattern pair at a higher resolution. This approximation provides a reasonable starting point for gradient descent search. Therefore, in the second step, we scale up the 11×11 solution to 13×13 and then refine the solution using gradient descent optimization. This scale-and-refine process is repeated until reaching a resolution of 33×33. Figure 4.6 shows the evolution of this pattern optimization from 11×11 to 33×33, from left to right. The far right aperture pair is our final optimized coded aperture pair for DFD.

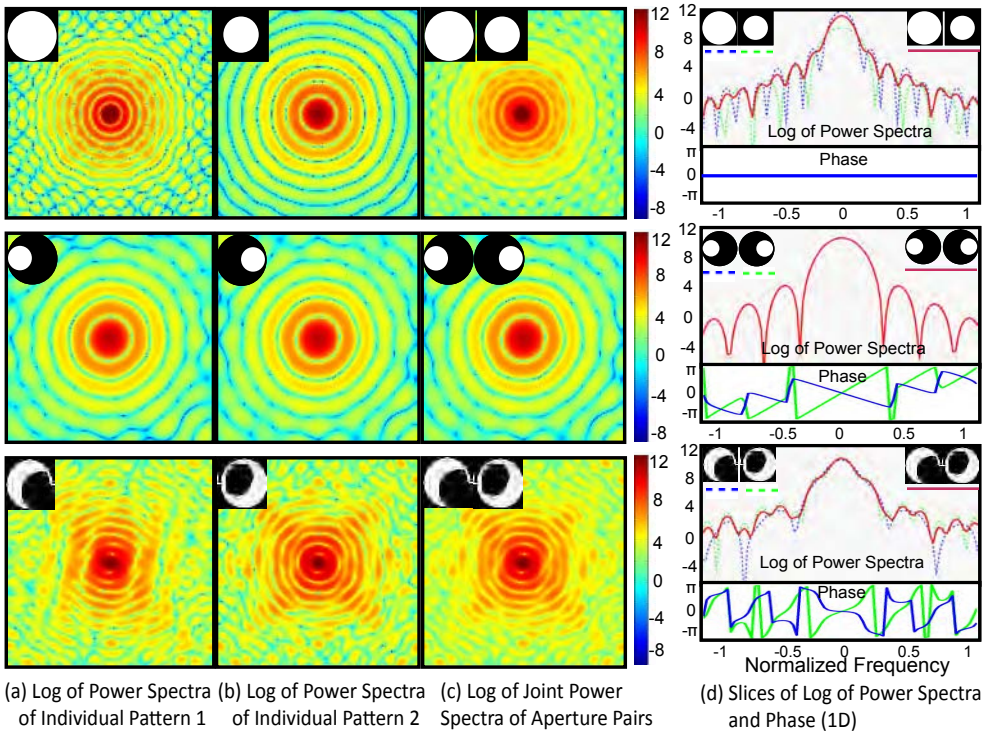


Figure 4.7: Pattern spectra of three different aperture pairs, including the optimized large/small circular aperture pair (Row 1), a pair of circular apertures with shifted centers (Row 2), and our optimized coded aperture pair (Row 3). The log of power spectra of each single pattern in the aperture pairs is illustrated in (a) and (b); and the log of joint power spectra of the aperture pairs is illustrated in (c). For a clearer illustration, one 1-D slice of each 2D power spectra is plotted in (d). In addition, one 1-D slice of phase of each single pattern is also plotted in (d). We can see the two patterns in the optimized coded aperture pair compensate each other in both power spectra and phase.

4.3.2 Discussion

4.3.2.1 On depth from defocus

The optimal radius ratio of a large/small aperture pair is shown to be 1.5 in Section 4.2.3. For an intuitive visualization of this ratio's optimality, we illustrate the large/small aperture pair with radius ratio 1.5 in the Fourier domain (Figure 4.7 (a, b), Row 1). One slice of the log of power spectrum of the large circular pattern ($log(|K_1|^2)$) is plotted as a dashed blue line in the first row of Figure 4.7 (d); the corresponding slice of the small circular pattern ($log(|K_2|^2)$) is plotted as a dashed green line. We can see that, due to the optimized ratio 1.5, these two power spectra compensate each other with respect to the zero-crossing frequencies. This compensation intuitively increases the relative defocus between the two PSFs and benefits the depth estimation.

One can also increase the relative defocus by designing a pair of patterns whose spectra compensate each other in phase. One example is a pair of small circular patterns with shifted centers (a stereo-like aperture pair) as shown in Figure 4.7, Row 2. These two patterns share the same power spectra, but compensate each other in phase (Figure 4.7 (d), Row 2). This compensation in phase yields a stereo-like effect in the captured images and increases the performance of DFD.

Remarkably, our optimized coded aperture pairs exhibit significant compensations in both power spectra and phase as shown in Figure 4.7 (d), Row 3. Intuitively, this compensation maximizes the score defined in Equation 4.9, greatly enhances the relative defocus, and improves the performance of DFD.

Figure 4.8 (a) shows the depth estimation curves $M(d,\, d^*, K_1,\, K_2)$ for the optimized circular aperture pair (green), a pair of shifted circular apertures (blue), and our optimized coded aperture pair (red). We can see the optimized coded aperture pair exhibits a more pronounced minimum in the profile of M than the other two pairs. This leads to depth estimation that is more precise and more robust to noise and scene variations.

Levin [87] further brought the idea of coded aperture pairs to a coded aperture set for depth estimation. In particular, their analysis gave an upper bound on the best possible depth discrimination from coded apertures, and verified that our optimized coded aperture pairs are near-optimal.

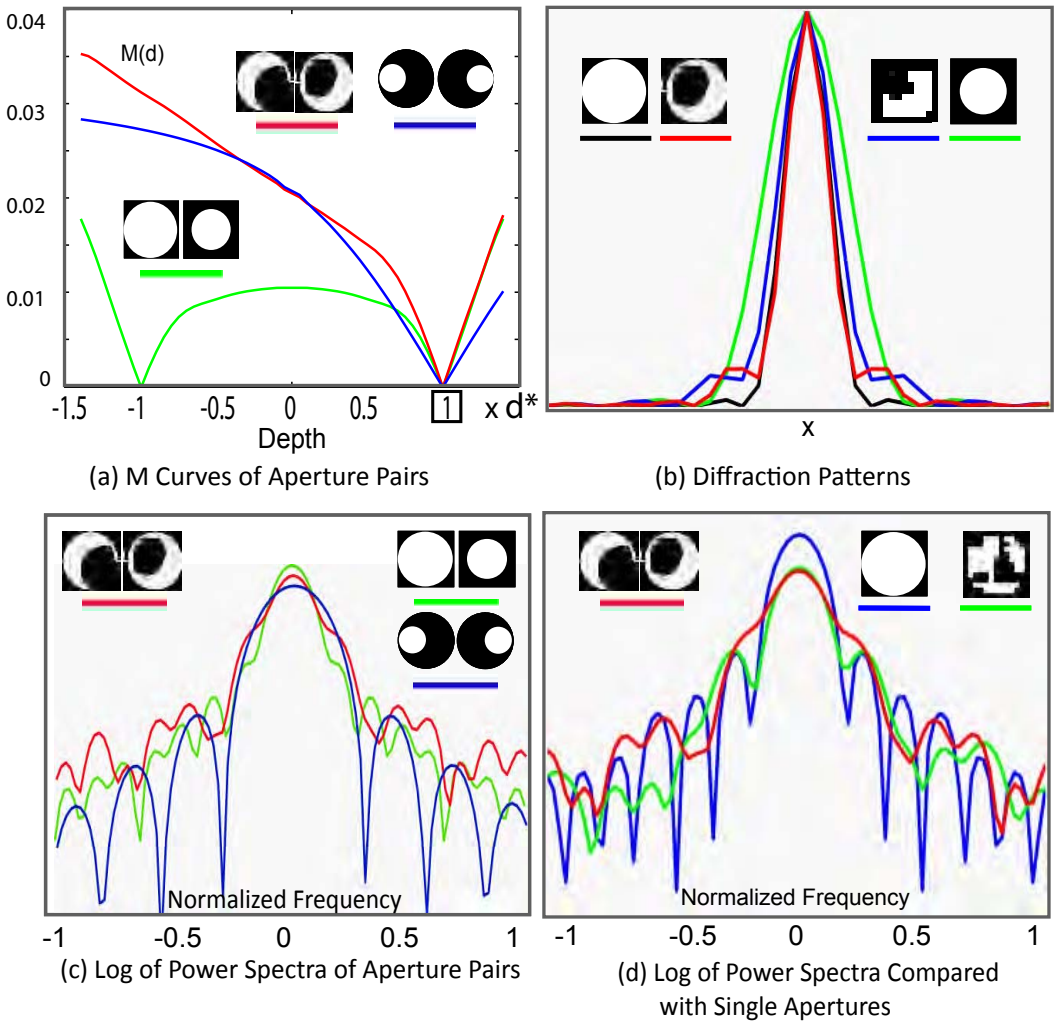


Figure 4.8: (a) Comparison of M curves among the optimized coded aperture pair, optimized circular aperture pair and the stereo-like aperture pair. (b) The in-focus diffraction patterns of four apertures, including a large circular aperture, a small circular aperture, one of our optimized coded apertures at high resolution, and one of our optimized coded aperture at low resolution. (c) Comparison of the joint power spectra of the optimized coded aperture pair with those of the other two aperture pairs. (d) Comparison of the joint power spectra of the optimized coded aperture pair with the power spectra of several single aperture patterns, including a conventional circular aperture and one coded aperture optimized for defocus deblurring in the previous chapter.

4.3.2.2 On defocus deblurring

Equation 4.3 implies broadband joint power spectra will bring great improvements in the quality of defocus deblurring. Although the aperture pairs are optimized for best DFD, the resulting complementary power spectra enable us to also compute a high quality all-focused image from the two captured defocused images. This is because, with zero-crossings located at different frequencies for each of the two apertures, the two apertures jointly provide broadband coverage of the frequency domain. Log of the joint power spectra of the aperture pairs $log(|K_1|^2/2 + |K_1|^2/2)$ are shown in Figure 4.7 (c). For the optimized circular aperture pair and the optimized coded aperture pair, the joint pattern pairs are much more broadband than the individual patterns. 1-D Slices of the power spectra of three single aperture patterns are shown in Figure 4.7 (d) for a clearer illustration.

Two defocused images with different blur kernels can be much better than each single image. This is an important implication of Equation 4.3. Rav-Acha and Peleg discussed a similar idea in the context of motion-blur deblurring [132], but do not provide detailed reasoning or a closed-form deblurring algorithm.

For the stereo-like pair with shifted circular patterns, its power spectra does not have any compensation one another. Its joint power spectra thus contains many zero-crossings as shown in Figure 4.7 (b), Row 2, and the aperture pair is therefore not ideal for defocus deblurring. The joint power spectra of the three aperture pairs are compared in Figure 4.8 (c).

4.3.2.3 On diffraction

The final optimized aperture pair of resolution 33×33 is not only superior to the solution at 11×11 in terms of the evaluation criterion defined in Equation (4.9), but also produces less diffraction because of greater smoothness in the pattern. In Figure 4.8 (c), the in-focus diffraction pattern of one of our optimized apertures is compared to three other aperture patterns, including a large circular aperture, a small circular aperture, and an optimized pattern at a lower resolution (the first pattern in Figure 4.6). We can see that the diffraction pattern of the optimized pattern at a high resolution is more compact than the small circular aperture and the optimized pattern at a low resolution.

4.4 Recovery of depth and all-focused Image

With the optimized aperture pair, we use a straightforward algorithm to estimate the depth map U and recover the latent all-focused image I. For each sampled depth value $d \in \mathcal{D}$, we compute $\hat{F_0}^{(d)}$ according to Equation (4.3) and then reconstruct two defocused images. At each pixel, the residual $W^{(d)}$ between the reconstructed images and the observed images gives a measure of how close d is to the actual depth d^*:

$$W^{(d)} = \sum_{i=1,2} |IFFT(\hat{F_0}^{(d)} * K_i^{\hat{d}} - F_i)|, \tag{4.12}$$

where IFFT is the 2D inverse Fourier transform. With our optimized aperture pairs, the value of $W^{(d)}(x, y)$ reaches an obvious minimum for pixel (x, y) if d is equal to the real depth. Then, we can obtain the depth map U as

$$U(x, y) = \arg\min_{d \in \mathcal{D}} W^{(d)}(x, y), \tag{4.13}$$

and then recover the all-focused image I as

$$I(x, y) = \hat{F_0}^{(U_{x,y})}(x, y). \tag{4.14}$$

The most computationally expensive operation in this algorithm is the inverse Fourier transform. Since it is $O(N \log(N))$, the overall computational complexity of recovering U and I is $O(l \cdot Nlog(N))$, where l is the number of sampled depth values and N is the number of image pixels. With this complexity, real-time performance is possible. In our Matlab implementation, this algorithm takes 15 seconds for a defocused image pair of size 1024×768 and 30 sampled depth values. Greater efficiency can be gained by simultaneously processing different portions of the image pair in multiple threads.

From the sparsely sampled depth values, we increase the depth resolution at a location (x, y) by fitting the sequence of residuals $\{W_{xy}^{(d-2)}, W_{xy}^{(d-1)}, W_{xy}^{(d)}, W_{xy}^{(d+1)}, W_{xy}^{(d+2)}\}$ with a 3^{rd}-order polynomial curve: $v = a_1 d^3 + a_2 d^2 + a_3 d + a_4$. With this interpolating polynomial, a continuous-valued depth estimate d' can be obtained from the curve's minimum ($\delta v / \delta d = 0$).

4.4.1 Performance analysis

To quantitatively evaluate the optimized coded aperture pair, we conducted experiments on a synthetic staircase scene with two textures, one with strong and dense patterns, and another of natural

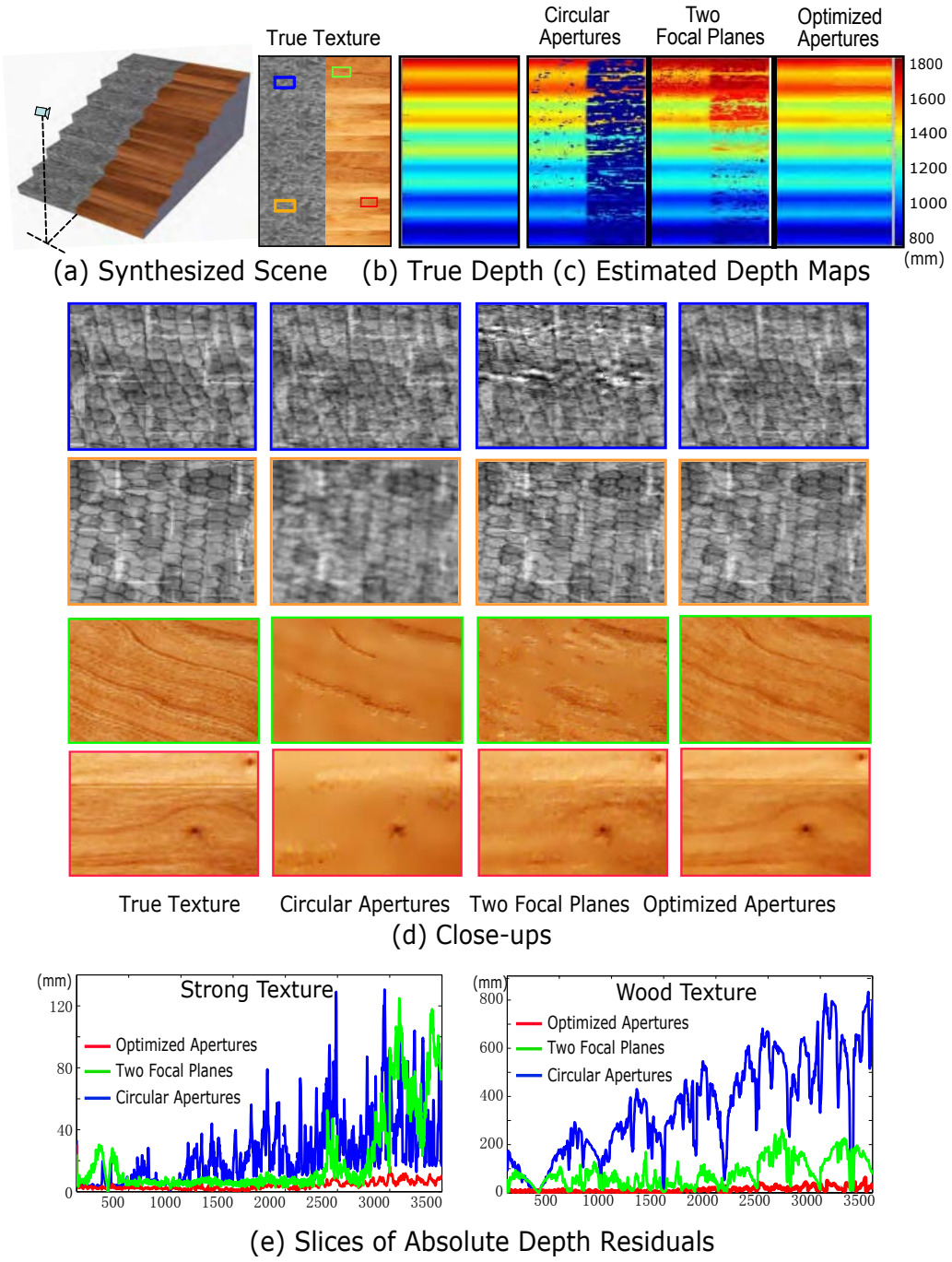


Figure 4.9: Comparison of depth from defocus and defocus deblurring using a synthetic scene. (a) 3-D structure of synthesized stairs and the groundtruth of texture map. (b) Groundtruth of the depth map. (c) Estimated depth maps using three different methods. From left to right: small/large circular aperture pair, two focal planes, and the proposed coded aperture pair. (d) Close-ups of four regions in the ground truth texture and the images recovered using the four different methods. (e) Left: The depth residuals of the four depth estimation methods on the strong texture; right: the depth residuals on the wood texture.

wood with weak texture. The virtual camera (focal length = 50mm, pixel size = 10 $\mu\ m$) is positioned with respect to the stairs as shown in Figure 4.9 (a). The corresponding ground truth texture and depth map are shown in (b) and (c), respectively. Comparisons are presented with two other typical aperture configurations: a small/large circular aperture pair, and a circular aperture with two sensor locations (shift of focus plane rather than change in aperture radius).

For the DFD algorithm using our optimized aperture pair, the focal plane is set near the average scene depth (1.2m) so that the maximum blur size at the nearest/farthest points is about 15 pixels. For the conventional method using a small/large circular aperture pair, the focal plane is set at the nearest scene point to avoid front/behind ambiguity with respect to the focal plane and yet capture the same depth range. This leads to a maximum blur size of about 30 pixels at the farthest point. The radius ratio of the two circular apertures is set to 1.5, the optimal value.

For the DFD method with two sensor positions, [143] reveals that moving the sensor in a DOF interval is optimal with respect to estimation robustness, and the depth estimation can be unstable if the interval is larger than the DOF by a factor of 2 or higher. However, in many scenes, including this simulated one, the depth range is often far larger than the DOF and therefore the optimal interval is practically not achievable. In this simulation, the two defocused images are synthesized with focal planes set at the nearest point (0.8m) and the farthest point (1.8m). Identical Gaussian noise ($\sigma = 0.005$) is added to all the synthesized images.

Figure 4.9 (d) shows results of the three DFD methods. Note that no post-processing is applied in this estimation. By comparing to (c), we can see that the depth precision of our proposed method is closest to the ground truth. For a clearer comparison, depth residuals are plotted in (f) for vertical slices of the computed depth maps, with the strong texture in the top plot and the wood texture at the bottom. At the same time, our proposed method generates an all-focused image of higher quality than the other two methods, as illustrated in (e).

A quantitative comparison among these dual-image DFD methods is given in Table 1. Using the optimized coded aperture pair leads to considerably lower root-mean-squared errors (RMSE) for both depth estimation and defocus deblurring in comparison to the conventional circular aperture pair and the two focal planes methods. The difference in performance is particularly large for the natural wood with weaker texture, which indicates greater robustness of the optimized pair.

Table 1. Quantitative evaluation of depth and deblurring error

	Strong Texture (RMSE)		Wood Texture (RMSE)	
	Depth (mm)	Grayscale	Depth (mm)	Color
Circular apertures	27.28	0.028	464.04	0.060
Two focal planes	6.32	0.027	124.21	0.045
Proposed coded apertures	4.03	0.016	18.82	0.036

4.5 Experiments with real apertures

(a) A disassembled Canon EF 50mm f/2.8 lens (b) Two lenses with the optimized patterns inserted

Figure 4.10: Implementation of aperture pair. (a) Lenses are opened. (b) Photomasks with the optimized aperture patterns are inserted.

We printed our optimized pair of aperture patterns on high resolution (1 micron) photomasks, and inserted them into two Canon EF 50mm $f/1.8$ lenses (See Figure (4.10)). These two lenses are mounted to a Canon EOS 20D camera in sequence to take a pair of images of each scene. The camera is firmly attached to a tripod and no camera parameter is changed during the capturing. Switching the lenses often introduces a displacement of around 5 pixels between the two captured images. We correct for this with an affine transformation.

Figure 4.11 shows a scene with large depth variation, ranging from 3 meters to about 15 meters. We intentionally set the focus to the nearest scene point so that the conventional DFD method, which uses a circular aperture, can be applied and compared against. For the conventional method, the f-Number was set to $f/2.8$ and $f/4.5$, respectively, such that the radius ratio is close to the optimal

value determined in Section 4. For a fair comparison, all of the four input images were captured with the same exposure time.

The results are similar to those from our simulation. We can see clearly from Figure 4.11(b) that depth estimation using the conventional circular apertures only works well in regions with strong texture or sharp edges. On the contrary, depth estimation with the optimized coded apertures is robust to scenes with subtle texture. Note that the same depth estimation algorithm as described in Section 5 is used here for both settings, and no post-processing of the depth map has been applied.

Figure 4.12 shows a scene inside a bookstore. The depth range is about 2-5 m. Two images (a,b) were taken using the optimized coded aperture pair with the focus set to 3m. The computed all-focused image and depth map are shown in (c) and (d). The ground truth images (e) were captured with a tiny aperture ($f/16$) and long exposure time. We can see that the computed all-focused image exhibits accurate deblurring over a large depth of field and appears very similar to the ground truth image.

Figure 4.13 (a) shows a scene ranging in depth from 3 to 8 meters. The focus is set to the middle of the depth of field. The scene shown in Figure 4.13 (b) has a depth range of 2 to 6 meters, with focus set to 2 meters. The computed depth maps and all-focused images of these two scenes are illustrated in the figure as well. Close-ups of four regions in these two scenes are shown in the bottom. Although the texture of most regions in these two scenes are quite weak, we can see that the estimated depth maps are still smooth and accurate. Note that we used a straightforward depth estimation algorithm as described in Section **??** and do not impose any smoothness constraint.

4.6 Summary

In this chapter, we answer the question 'What are good PSFs for depth from defocus' by presenting a comprehensive criterion for evaluating aperture patterns for the purpose of DFD. This criterion is used to solve for an optimized pair of apertures that complement each other both for estimating relative defocus and for preserving frequency content. This optimized aperture pair enables more robust depth estimation in the presence of image noise and weak texture. This improved depth map is then used to deconvolve the two captured images, in which frequency content has been well preserved, and yields a high-quality all-focused image.

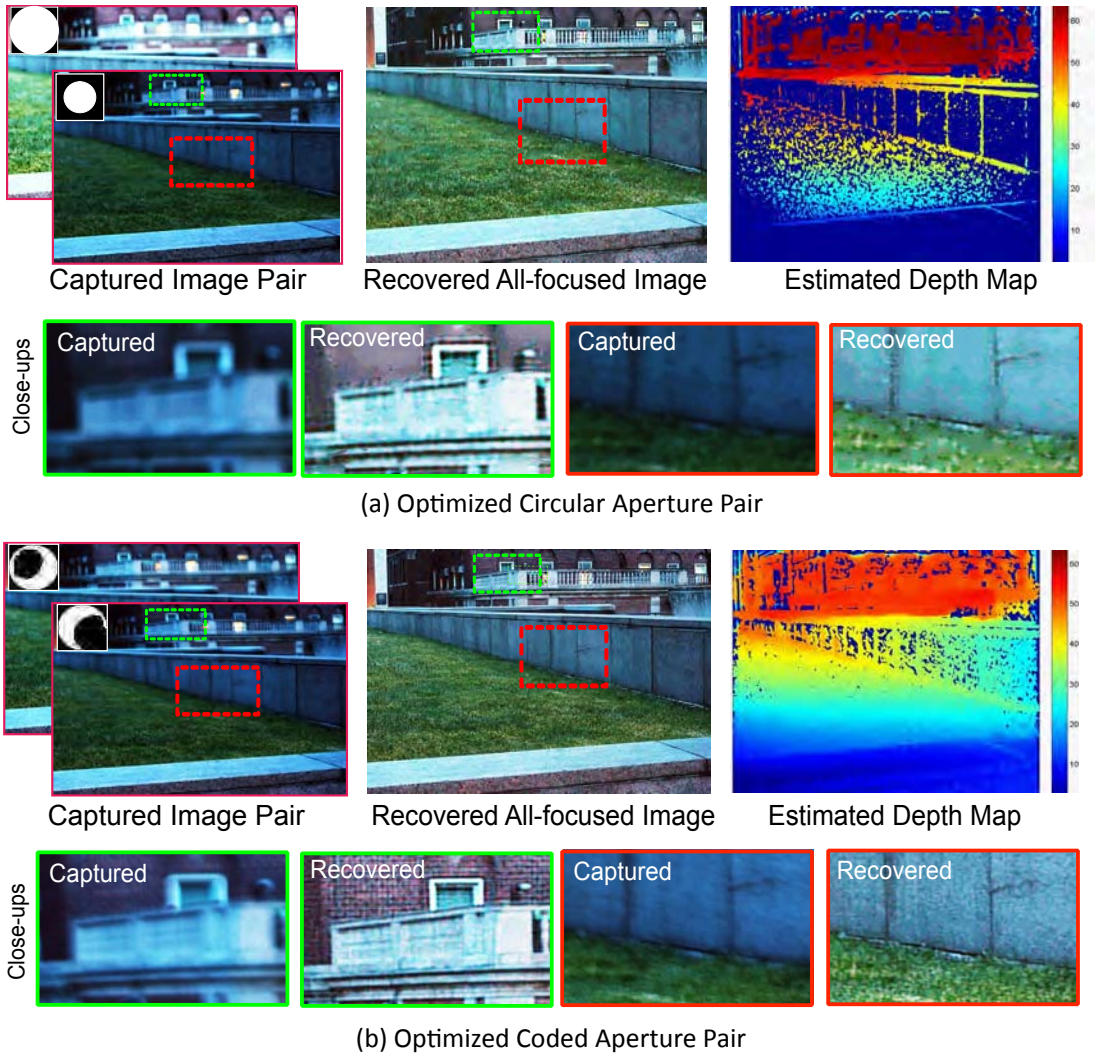


Figure 4.11: Campus view. (a) Conventional DFD method using circular apertures of different size. The two input images are captured with $f/2.8$ and $f/4.5$, respectively. (b) DFD method using the optimized coded aperture pair. All the images are captured with focus set to the nearest point. Note that the only difference between (a) and (b) is the choice of the aperture patterns.

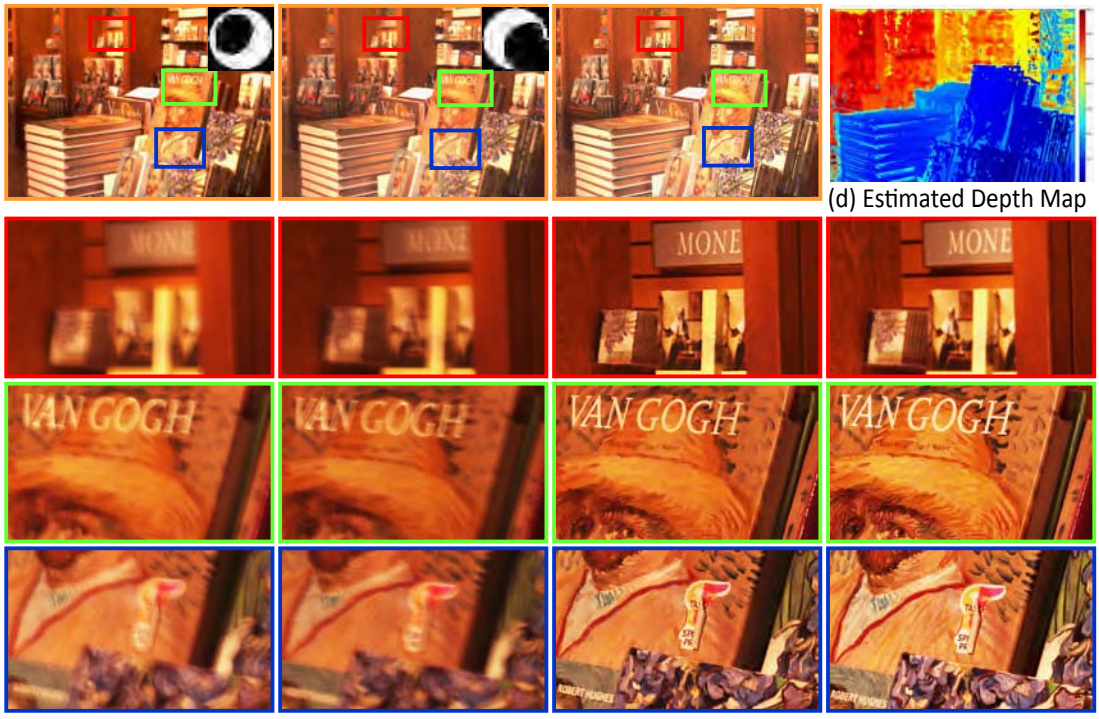


Figure 4.12: Inside a book store. (a-b) Captured Images using the coded aperture pair with close-ups of several regions. The focus is set at the middle of depth of field. (c) The recovered image with close-ups of the corresponding regions. (d) The estimated depth map without post-processing. (e) Close-ups of the regions in the ground truth image which was captured by using a small aperture $f/16$ and a long exposure time.

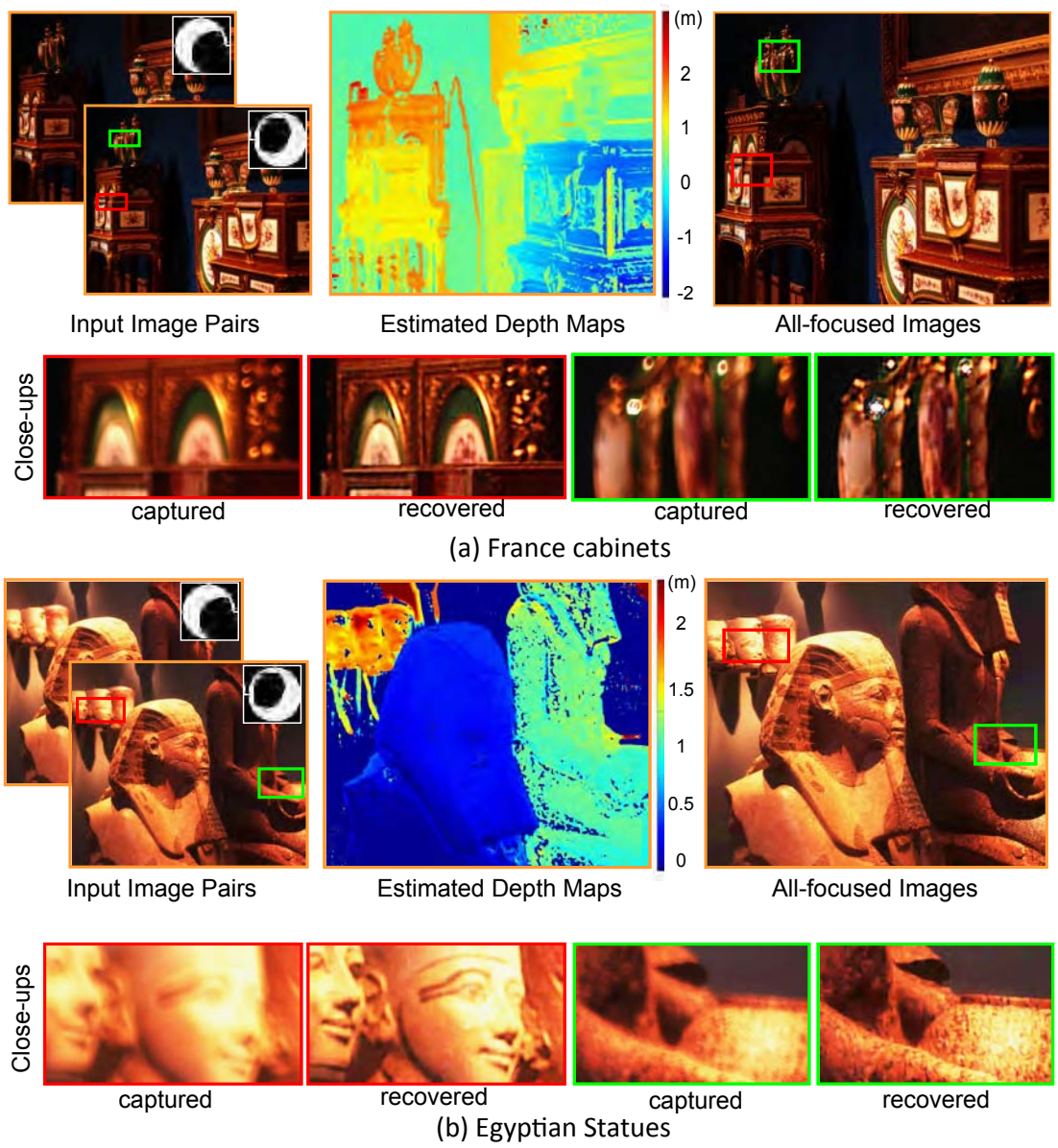


Figure 4.13: France cabinets and Egyptian statues. (a) France cabinets: captured image pairs using the coded aperture pair with focus set to the middle of the depth of field. (b) Egyptian statues: captured image pairs using the coded aperture pair with focus set to the nearest point. The blur size of objects with no texture are automatically set to 0.

Chapter 5

Depth from diffusion

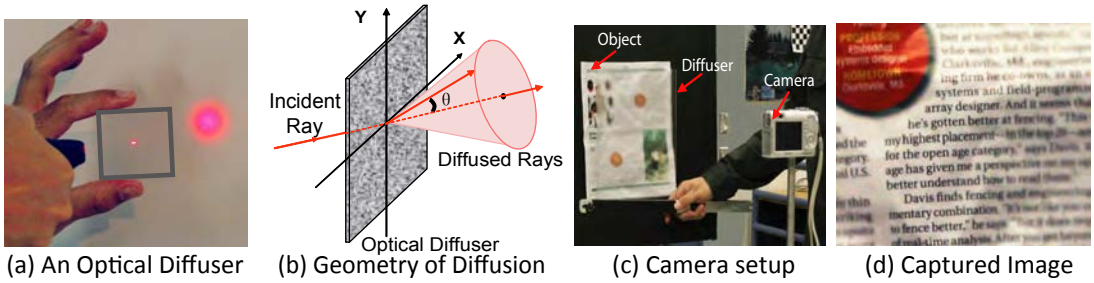
5.1 Introduction

In DFD, depth information of the scene is encoded in the PSF scale. While aperture coding optimizes the PSF and helps to improve the precision of DFD, the depth sensitivity is rigidly limited by aperture size. Schechner and Kiryati [144] shows that DFD can be regarded as a triangulation-based method. The aperture size in DFD plays the same role as the baseline B in stereo vision. We can thus apply the depth sensitivity analysis used in stereo vision to a DFD system as follows:

$$S \approx m \cdot B/U = D \cdot m/U, \tag{5.1}$$

where D is the aperture diameter. For any given magnification m, the sensitivity is proportional to the aperture size D and inversely proportional to the distance U. To transcend this fundamental limit, we propose a novel depth recovery technique using an optical diffuser – referred to as depth from diffusion (DFDiff).

In optics, a diffuser is a device that diffuses (or scatters) light and is widely used to soften or shape light in illumination or display[102][98]. Optical diffusers are also commonly used in commercial photography. Photographers place diffusers in front of the flash to get rid of harsh light, in front of the lens to soften the image, or at the focal plane to preview the image. Most commercially available diffusers are implemented as a refractive element with a random surface profile. These surfaces can be created using random physical processes such as sandblasting and holographic exposure, or programmatically using a lithographic or direct writing method [15][152][57][30]. Figure



(a) An Optical Diffuser (b) Geometry of Diffusion (c) Camera setup (d) Captured Image

Figure 5.1: (a) A laser beam is diffused by a holographic diffuser. (b) The geometry of the optical diffusion. (c) An optical diffuser is placed in front of the camera and close to the object (a crinkled magazine). (d) A close-up of the captured image. We can see that the blur of the text is spatially varying as a function of depth.

5.1(a) shows an off-the-shelf diffuser scattering a beam of light.

A diffuser converts an incident ray into a cluster of scattered rays. This behavior is fundamentally different from most conventional optical devices used in imaging, such as mirrors and lenses. Figure 5.2(b) illustrates the geometry of light scattering in Figure 5.1(a). The scattering properties of a diffuser can be generally characterized by its diffusion function $\mathcal{D}(\theta_i, \psi_i, \theta_o, \psi_o)$, where $[\theta_i, \psi_i]$ is the incident direction and $[\theta_e, \psi_e]$ is the exitance direction. Since diffusers have usually been designed so that scatter is invariant to incident direction, the diffusion function can be simply written as $\mathcal{D}(\theta, \psi)$, where θ and ψ are the angular coordinates of the exiting ray relative to the incident direction. For most commercial diffusers (e.g., the one shown in Figure 5.1), the diffusion functions are radially symmetric and can be further simplified to $\mathcal{D}(\theta)$.

We analyze how optical diffusers affect image formation when they are present in an imaging system. When a diffuser is placed in front of the objects, we capture diffused (or blurred) images which have similar appearance as defocused images. By assuming locally constant diffusion, a small diffused image can be formulated as the convolution of the clear image and the diffusion kernel, whose shape is decided by the diffusion pattern of the diffuser and whose size relies on the distance from the object to the diffuser. This is mathematically identical to the well-known lens defocus, which is often formulated as the convolution of an in-focus image and the defocus kernel. This analogy enables us to reuse the previously proposed PSF evaluation criterion for DFDiff. The main benefit of DFDiff is that while DFD requires very large apertures to improve depth sensitivity,

DFDiff only requires an increase in the diffusion angle – a much less expensive proposition.

To implement our depth from diffusion (DFDiff) technique, we place an optical diffuser between the scene and the camera as shown in Figure 5.1(c). Our analysis shows that the diffusion blur size is proportional to the object-to-diffuser distance (see Figure 5.1(d)). We can therefore infer depth by estimating the diffusion blur size at all points in the image. Since the depth estimation problem for DFDiff is similar to conventional DFD, many existing algorithms can be used to find a solution.

While DFDiff is similar in principle to DFD, it offers three significant advantages:

- **High-precision depth estimation with a small lens.** For DFDiff, the precision of depth estimation depends only on the mean scattering angle of the diffuser and is independent of lens size. Note that while it is often difficult to make lenses with large apertures, it is relatively easy to make diffusers with large diffusion angles.

- **Depth estimation for distant objects.** By choosing the proper diffuser, DFDiff can achieve high precision depth estimation even for objects at very large distances from the camera. For DFD, depth sensitivity is inversely proportional to the square of object distance [37][13]. In many scenarios, it is necessary to place objects far from the camera in order to achieve a reasonable field of view.

- **Less sensitive to lens aberrations.** Lens aberrations cause the shape of the defocus point spread function (PSF) to vary with field position. This effect is strong, particularly in the case of inexpensive lenses, and degrades the precision of depth estimation. In contrast, as we show in Section 6, diffusion PSFs are more invariant to field position.

DFDiff does, however, require the flexibility to place a diffuser in the scene, which is impractical or impossible in some situations.

5.2 Image formation with a diffuser

5.2.1 Geometry of diffusion

When an optical diffuser is placed between the scene and the camera, the captured image will be diffused, or blurred. The diffusion varies with camera, scene, and diffuser settings. We first show

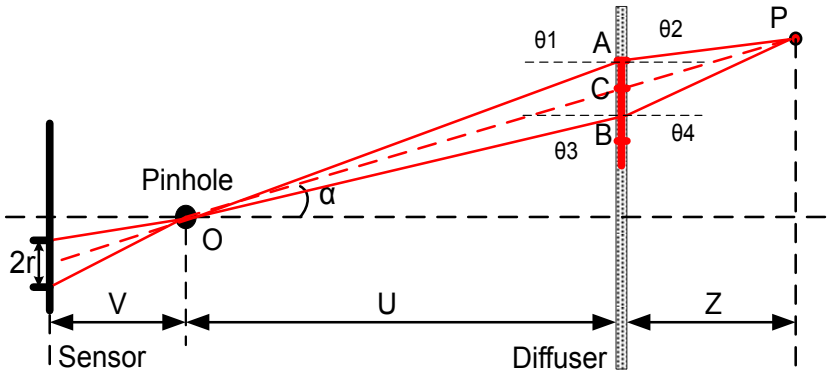


Figure 5.2: Geometry of diffusion in a pinhole camera. An optical diffuser with a pillbox diffusion function of degree θ is placed in front of a scene point P and perpendicular to the optical axis. From the viewpoint of pinhole, a diffused pattern AB appears on the diffuser plane.

in Figure 5.2 the geometry of diffusion in a simple pinhole imaging system. Placed between the pinhole O and the scene point P is a diffuser with a pillbox diffusion function $\sqcap_\theta(x)$:

$$\sqcap_\theta(x) = \begin{cases} \frac{1}{\pi \cdot \theta^2} & x < \theta \\ 0 & otherwise, \end{cases}$$

where θ is the diffusion angle of the diffuser.

As shown in Figure 5.2, the light from an arbitrary scene point P is scattered by the diffuser. Due to the limit of the diffusion angle θ, only the light scattered from a specific region AB can reach the pinhole O. From the viewpoint of the pinhole, a line AB (or pillbox in 2D) appears on the diffuser plane instead of the actual point P.

Proposition 5.2.1 *When an optical diffuser is placed parallel to the sensor plane (see Figure 5.2) and the diffusion angle θ is small ($\sin\theta \approx \theta$), we get*

$$\frac{2\tan\theta}{\cos^2\alpha} \cdot \frac{1}{\overline{AB}} = \frac{1}{U} + \frac{1}{Z}, \tag{5.2}$$

where α is the field angle and $\overline{AB}$ is the diffusion size. The perspective projection of P on the diffuser plane C can be approximated with high precision as the center of AB when α is not too large. (see Appendix E for the proof.)

This equation shows that for any given U, the diffusion size $\overline{AB}$ is uniquely determined by the

distance Z and the diffusion angle θ. In addition, the perspective projection C and the center of the diffusion pattern AB are the same. Therefore, the diffuser blur does not cause geometric distortions.

Then, the radius r of the PSF can be obtained using Equation 5.2:

$$r = \frac{V}{U} \cdot \frac{\overline{AB}}{2} = m \cdot \frac{Z}{\cos^2 \alpha} \cdot \tan \theta, \tag{5.3}$$

where $m = V/(Z + U)$ is the image magnification.

In this chapter we assume the diffuser is parallel to the sensor plane. The equations governing DFDiff can easily be extended to include tilted planes. Please see Appendix in the supplementary material for details.

5.2.2 Equi-diffusion surfaces and image formation

From Equation 5.3, we can see that the diffusion size r is related to the field angle α. Given r, we can derive a surface using Equation 5.3:

$$Z = \frac{r \cdot U \cdot \cos^2 \alpha}{\tan \theta \cdot V - r \cdot \cos^2 \alpha}, \tag{5.4}$$

referred to as an equi-diffusion surface. All scene points on an equi-diffusion surface will be equally blurred by diffusion. Under the paraxial approximation ($\sin \alpha = \alpha$), the surface is planar, since the term $\cos^2 \alpha$ approaches 1. For a large field of view, the equi-diffusion surface is no longer planar. A set of equi-diffusion surfaces in 1D space are shown in Figure 5.3.

For any equi-diffusion surface with $r = r_0$, the diffused image F can be written as the convolution of the latent clear pinhole image F_0 and the pillbox PSF $\sqcap_{r_0}$: $F = F_0 \otimes \sqcap_{r_0}$. Similarly, when a diffuser with Gaussian diffusion function is used, we will have $F = F_0 \otimes g_{r_0}$, where g_{r_0} is a Gaussian function with standard deviation $\sigma = r_0$. More generally, for a diffuser with an arbitrary diffusion function $\mathcal{D}$, the image formation can be written as the convolution of the image F_0 and the diffusion function $\mathcal{D}$ of size r_0:

$$F = F_0 \otimes \mathcal{D}_{r_0}. \tag{5.5}$$

5.2.3 Diffusion + Defocus

It is well known that for a lens camera without a diffuser, the defocused image of a fronto-planar object can be formulated as $F = F_0 \otimes L$, where F_0 is the latent focused image (pinhole image)

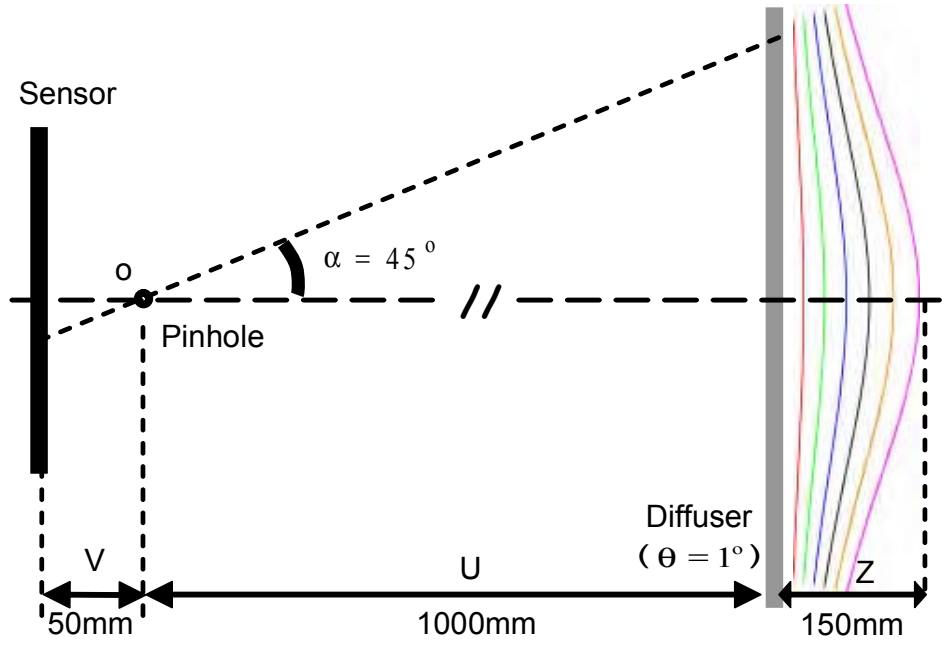


Figure 5.3: Equi-diffusion surfaces of a simulated pinhole camera with a diffuser. Six equi-diffusion surfaces (1D) are shown in different colors.

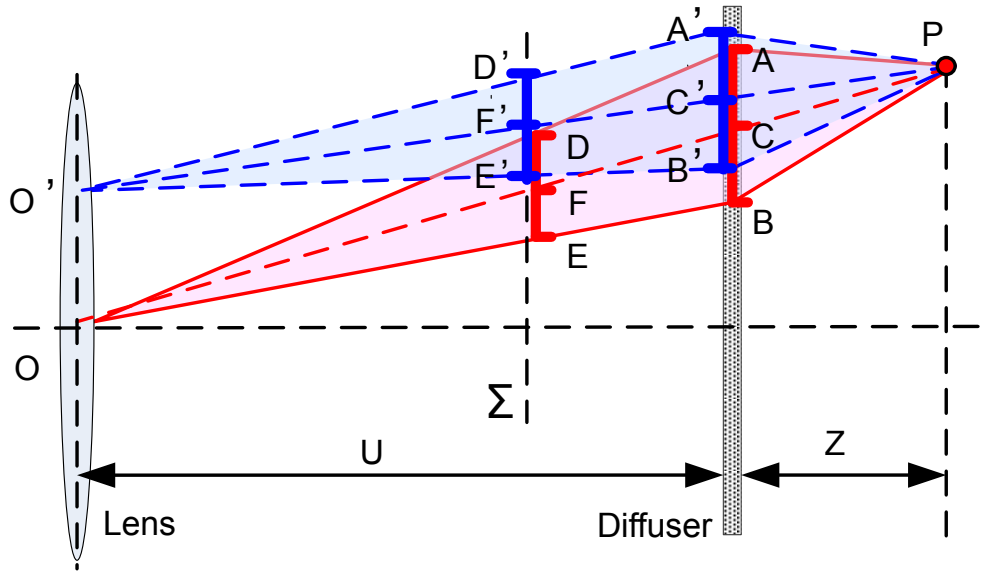


Figure 5.4: Diffusion in a lens camera. An optical diffuser with a pillbox diffusion function of degree θ is placed in front of a pinhole camera and perpendicular to the optical axis.

and L is the defocus PSF. On the other hand, we know from Section 5.2.2 that for a pinhole camera augmented by a diffuser, the image of an equi-diffusion surface can be written as $F = F_0 \otimes \mathcal{D}$, where $\mathcal{D}$ is the diffusion PSF. But how will the lens blur interact with the diffuser blur when a diffuser is used in a lens camera?

Proposition 5.2.2 *Suppose a lens camera is focused at an arbitrary distance, and an optical diffuser, which is parallel to the lens, is placed between the lens and a scene point P. When the distance from P to the lens plane is much larger than the aperture size, we have*

$$\mathcal{K} = \mathcal{L} \otimes \mathcal{D}, \tag{5.6}$$

where K is the image of P (the PSF), L is the image of P that would be captured if the diffuser were removed (the defocus PSF), and D is the image of P that would be captured if a pinhole were used instead of the lens (the diffusion PSF).

Proof: As shown in Figure 5.4, suppose the lens is focused at Plane Σ and the diffuser is placed at a distance U, perpendicular to the optical axis, and a scene point P is located behind the diffuser at a distance Z. From Section 5.2.1 we know that from the perspective of O, a scene point P appears as AB in the diffuser plane, or as DE on the focus plane Σ. The image of DE on the sensor is $\mathcal{D}$, the diffusion PSF of P if a pinhole camera were used. Similarly, for an arbitrary point O', P appears as $A'B'$ on the diffuser plane and $D'E'$ on the focus plane. Since $U + Z \gg O'O$, the view angles of P with respect to O and O' can be regarded as equal, thus $AB = A'B'$ and $DE = D'E'$. Therefore, the image of $D'E$ on the sensor is a shifted version of $\mathcal{D}$.

For an arbitrary O', the center of the virtual image F' is the projection of P on the focus plane. Note that this effect is independent of the diffuser properties. When all the points on the aperture are considered, each point forms a virtual image of P on the focus plane, whose image on the sensor is the lens defocus pattern L. Hence, the image of P on the sensor K is the sum of a set of shifted $\mathcal{D}$'s whose centers are given by $\mathcal{L}$. That is $\mathcal{K} = \mathcal{L} \otimes \mathcal{D}$.

Now, suppose we have two images of a scene captured using a normal lens, one without a diffuser and one with a diffuser placed in front of the object, as illustrated in Figure 5.4. Consider arbitrary corresponding small patches P_1 and P_2 in the two images. By assuming that the diffusion and defocus are locally constant, we have $P_2 = P_1 \otimes \mathcal{D}$, since $P_2 = P_0 \otimes (\mathcal{L} \otimes \mathcal{D})$ and $P_1 = P_0 \otimes \mathcal{L}$, where P_0 is the latent focused patch. According to Equation 5.2, $\mathcal{D}$ is determined by the diffusion

profile of the diffuser and the distance from the patch to the diffuser plane. Note that according to Proposition 5.2.2, this relation holds regardless of the lens focus.

5.3 Depth from diffusion algorithm

The basic idea of depth from diffusion (DFDiff) is straightforward. As shown in Figure **??**(a), an optical diffuser is placed between the scene and the camera, and a blurred image is captured (shown in Figure **??**(b)). The diffusion size is uniquely determined by the distance between objects and the diffuser. By estimating the diffusion size in the image, we can infer the scene depth relative to the diffuser plane.

To estimate the diffusion size, we can take two images F_1 and F_2 with and without a diffuser, respectively. According to Section 5.2.2, for an arbitrary small patch pair P_1 and P_2 in these images, we have $P_2 = P_1 \otimes \mathcal{D}_{s_0}$, where s_0 is the diffusion size. To estimate depth, we must infer the diffusion size s_0 from the two captured patches P_2 and P_1. Note that this is exactly the same formulation as conventional DFD, which computes depth from two input images, one defocused and one focused. Therefore, most existing DFD algorithms can be applied to estimate the diffusion size s_0. For complicated scene surfaces, different diffusion sizes have to be computed for different pixels. The same problem also exists in DFD and many strategies have been proposed to estimate maps of blur size.

In our implementation, we adapt a straightforward algorithm, similar to those in the previous chapter on DFD, to recover the map of diffusion size, S(x, y). For every sampled diffusion size s, a residual map R^s is computed as

$$R^s(x, y) = |F_1(x, y) \otimes \mathcal{D}_s(x, y) - F_2|. \tag{5.7}$$

Then, for each pixel (x, y), its diffusion size $S(x, y)$ is selected to minimize the corresponding residual:

$$S(x, y) = \arg\min_s R^s(x, y). \tag{5.8}$$

Based on the estimated diffusion map $S(x, y)$, we can then compute the depth map $Z(x, y)$ according to Equation 5.2. Note that the field angle α can be computed directly from the pixel position (x, y) and camera parameters, so that it is straightforward to convert between $S(x, y)$ and $Z(x, y)$.

5.3.1 Reflections from diffuser surface

Although the light transmission efficiency of diffusers can be quite high (92% for the Luminit holographic diffusers, which will be used in our experiments), some light is still directly reflected by the diffuser surface to the camera. Thanks to its extremely rough surface, light reflected from the diffuser is usually quite uniform. Therefore, its contribution to the captured image can be approximately modeled as $F = a * F' + b$, where F is the actual diffused image captured with reflections, F' is the ideal diffused image captured without any reflection, and a and b are two constants mainly determined by the light transmission efficiency of the diffuser.

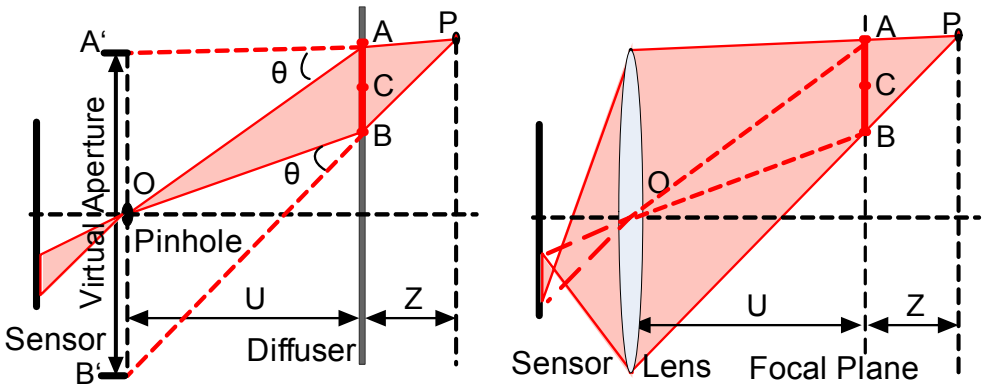
Obviously, for the mean brightness $\bar{F}$ and $\bar{F}'$, $\bar{F} = a * \bar{F}' + b$ still holds. $\bar{F}'$ can be estimated using the mean brightness B of the image captured without a diffuser. In addition, note that for a captured RGB image, $[a, b]$ is consistent over the three color channels. Therefore, given one image captured with a diffuser and one image captured without a diffuser, we can easily compute a and b by solving a simple linear equation. Then, the effects of reflectance can be removed by applying $F' = (F - b)/a$.

5.3.2 Illumination changes due to the diffuser

When a diffuser is placed over the object, the illumination will be first diffused by the diffuser before reaching the object. Illumination is usually low-frequency and the diffusion makes it even more uniform. Furthermore, non-specular surfaces are known to low-pass filter incident illumination. Therefore, illumination changes due to the diffuser will only affect low-frequencies in the captured images. To account for this effect, we apply a high-pass filter to Equation 5.7 and get

$$R^s(x, y) = \left| \mathbb{H}[F_1(x, y) \otimes D_s(x, y) - F_2] \right|, \tag{5.9}$$

where $\mathbb{H}$ is a high-pass filter. We use a Derivative of Gaussians (DOG) filter in our implementation. Note that the depth estimation mainly relies on the high-frequency information, so that applying a high-pass filter has little effect on depth estimation performance.



(a) Diffusion (b) Defocus

Figure 5.5: Equivalence between diffusion and lens defocus. The diffusion (a) caused by a diffuser in a pinhole camera is equivalent to the defocus (b) in a regular lens camera which has a large lens of size $A'B'$ and is focused at the diffuser plane.

5.4 Analysis

5.4.1 Diffusion vs. lens defocus

Diffusion caused by a diffuser can be shown to be geometrically equivalent to lens defocus. Figure 5.5(a) shows a pinhole camera with a diffuser placed in front of the scene point P, perpendicular to the optical axis. From the perspective of P, the pinhole O appears like a large aperture $A'B'$ which collects a cone $A'PB'$ of light from P. It should be noted that if we replace the pinhole with a lens of size $A'B'$, set the focus at the diffuser plane, and remove the diffuser as shown in (b), P will have the same projection AB on the focus plane, mapping to the same PSF on the sensor plane.

From Figure 5.5(a), we can see the size of the virtual aperture $A'B' = \frac{U+Z}{Z} \cdot AB$. We can compute AB from Equation 5.5, giving

$$A'B' = 2\tan\theta \cdot U/\cos^2\alpha. \tag{5.10}$$

When α is small, $A'B' = 2\tan\theta \cdot U$. For instance, a DFDiff system which consists of a pinhole camera and a 5° pillbox diffuser placed $1m$ away is equivalent to a DFD system whose lens has a huge aperture (diameter$= 17.5cm$) and is focused at $1m$.

While it is often expensive or even impossible to manufacture large lenses, it is relatively easy to make large diffusers with large diffusion angles. Several companies now supply off-the-shelf optical

diffusers with diffusion angles ranging from 0.2° to 80°. Because a diffuser effectively increases the lens aperture without physically increasing lens size, DFDiff provides an economical alternative for applications that require high precision in depth estimation.

5.4.2 Depth sensitivity

In depth from stereo, the disparity r is used to compute the depth Z [13][37][9]. The derivative of r with respect to Z, is often referred to as depth sensitivity $S = \partial r/\partial Z$. Usually, we have $S \approx B \cdot V/U^2 = m \cdot B/U$, where B is the baseline, U is the distance to the object, V is the distance from the lens to the sensor, and m is the image magnification. The higher the depth sensitivity is, the more precise is the depth estimation. As mentioned earlier, the depth sensitivity of a DFD system can be computed as $S \approx m \cdot B/U = m \cdot D/U$ (Equation 5.1).

A DFDiff system is equivalent to a DFD system with aperture size $D \approx 2U \cdot \tan\theta$ (Equation 5.10) when α is small. Therefore, we have $S \approx m \cdot 2\tan\theta$, where θ is the diffusion angle of the diffuser. For any given magnification m, the sensitivity only relies on θ.

To increase the depth sensitivity with DFD, one has to either increase the aperture size of the lens, which may be prohibitively expensive, or move the camera closer to the object, which reduces the field of view (FOV). However, for DFDiff, it is easy to achieve high depth precision at a large distance, even with a low-end lens.

5.4.3 Sensitivity, distance, and field of view

Suppose we have a Canon EOS 20D D-SLR camera, whose sensor has a dimension of $22.5mm \times 15mm$ 8 microns pixel size, and we have a target object of size $225mm \times 150mm$. Table 5.1 shows the required F# or Aperture diameter, D, in DFD, and the required diffusion angle θ in DFDiff for different depth precision requirements ($10\ pixel/mm, 1\ pixel/mm, 0.1\ pixel/mm$) and object distances ($500mm, 1000mm, 5000mm$). To ensure that the field of view (FOV) covers the whole object, the effective focal length (EFL) is increased with object distance. For example, the first row shows that if a depth precision of $10\ pixel/mm$ is required, for an object placed $500mm$ from the camera, then DFD requires a lens with $EFL = 50mm$ and F# = 0.125 ($D = 400mm$). DFDiff, on the other hand, can estimate depth with the same precision using any lens when a 21.80° diffuser is used.

FOV	U	S	EFL	DFD		DFDiff
mm×mm	*mm*	*pixel/mm*	*mm*	*F#*	*D (mm)*	θ
225×150	500	10	50	**0.125**	**400**	21.80°
225×150	500	1	50	1.25	40	2.29°
225×150	500	0.1	50	12.5	4	0.23°
225×150	1000	10	100	**0.125**	**800**	21.80°
225×150	1000	1	100	**1.25**	**80**	2.29°
225×150	1000	0.1	100	12.5	8	0.23°
225×150	5000	10	500	**0.125**	**4000**	21.80°
225×150	5000	1	500	**1.25**	**400**	2.29°
225×150	5000	0.1	500	12.5	40	0.23°

Table 5.1: Comparison of DFD and DFDiff for different depth precision requirements and object distances. On the left are FOV, object distance, and depth sensitivity that we want to achieve; on the right are the required EFL, F# or aperture size D in DFD and diffusion angle θ in DFDiff. In bold are lenses required by DFD which are too complicated to manufacture (e.g. a $500mm$ focal length lens with $4m$ diameter aperture).

We can see that for high precision and large object distance requirements, DFD demands lenses with unreasonably large apertures (e.g. a $500mm$ focal length lens with $4m$ diameter aperture). These lenses are shown in bold. DFDiff, on the other hand, can estimate high-precision depth maps using lenses with small apertures.

5.5 Experiments

Today, several companies sell off-the-shelf diffusers reproduced onto glass or plastic sheets up to 36" wide. In our experiments, we use holographic diffusers with Gaussian diffusion functions from Luminit Optics. These diffusers have different diffusion angles, ranging from $0.5°$ to $20°$, and different sizes, ranging from $2'' \times 2''$ to $10'' \times 8''$. In each experiment, the proper diffuser was chosen according to the scene and precision requirements.

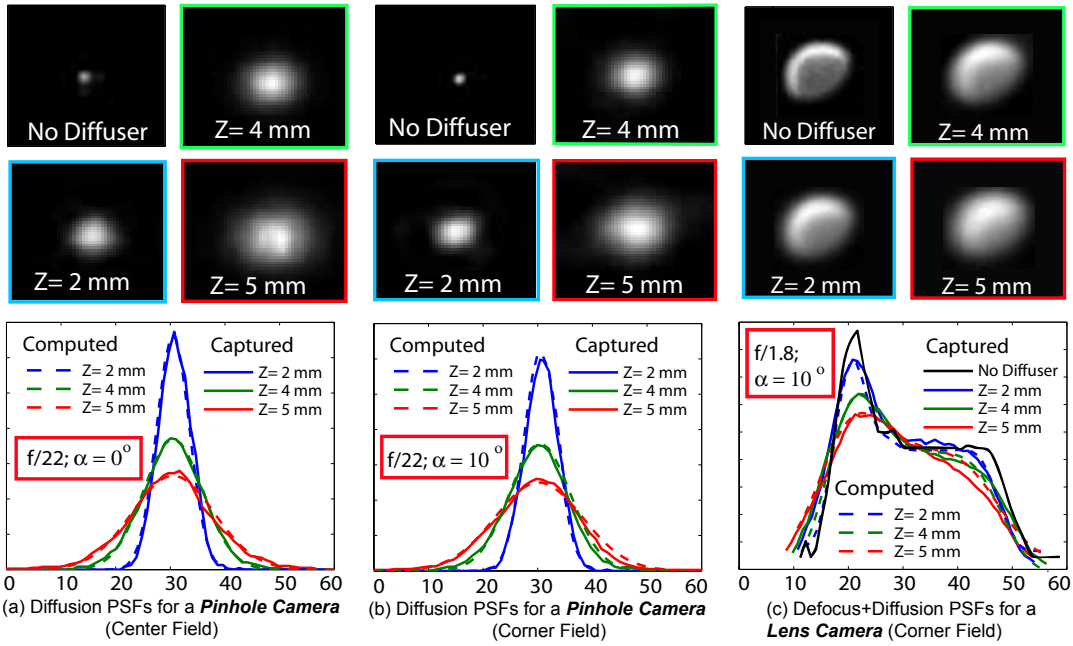


Figure 5.6: Model Verification. (a) Captured and computed diffusion PSFs of a center point source in a pinhole camera. (b) Captured and computed diffusion PSFs of a corner point source ($\alpha = 10°$) in a pinhole camera. (c) Captured and computed diffusion defocus+diffusion PSFs of a corner point source ($\alpha = 10°$). We can see that in all these three cases, the PSFs computed using our derived diffuser model (dashed curves) are fairly consistent with the captured ones (solid curves). Note that the defocus pattern in (c) is asymmetric because of lens aberrations.

5.5.1 Model verification

5.5.1.1 Pinhole camera

We first conducted experiments to verify the image formation model derived in Section 5.2. An array of point light sources was placed $1m$ in front of a Canon EOS T1i D-SLR camera with a Canon EF $50mm$ F/1.8 lens, perpendicular to the optical axis. First, to emulate a pinhole camera, we stopped down the aperture size to F/22. We mounted a $10°$ Luminit diffuser to a high-precision positioning stage, placing it just in front of the point light source array. We then captured a set of images while slowly moving the diffuser away from the light source array ($Z = 2mm - 10mm$).

Figure 5.6(a) left shows a focused image of the center point light source captured without a

diffuser. On the right we show three images captured with a diffuser placed at different positions ($2mm, 4mm$, and $5mm$). These three blurred images should be a convolution between the focused image and the three corresponding diffusion PSFs. Cross sections of the blurred images are plotted in solid curves on the right of Figure 5.6(a). Since the diffusion function of the diffuser and the distances Z are known, we can compute the diffusion PSFs according to our proposed imaging model. We then compute three diffused images by convolving these computed PSFs with the focused image. These three computed images are plotted in dashed curves. Figure 5.6(b) shows the captured images of a point light at the corner field ($\alpha = 10^{\circ}$), as well as a comparison with the computed images.

We can see from both Figure 5.6 (a) and (b) that the computed images are quite consistent with the captured ones. This indicates the real diffusion PSFs not only fit the designed patterns well, but also are spatially invariant.

5.5.1.2 Lens camera

To verify the proposed imaging model in the presence of defocus, we open up the aperture of the lens to F/1.8, focus the camera at a distance of $1.9m$, and repeat the same experiment as in Section 5.5.1. Figure 5.6(c) left shows the defocused image of a corner point source ($\alpha = 10^{\circ}$) captured without a diffuser. On the right we show three diffused and defocused images that were captured with the diffuser placed at different depths. We computed the diffusion PSFs from our diffusion model and convolved them with the defocused image captured without a diffuser. The computed diffused and defocused images are plotted in Figure 5.6(c) (dashed curves).

In Figure 5.6(b), note that although the aperture pattern of this Canon lens is circular, the captured defocus pattern is not circular at the periphery of the FOV, due to lens aberrations. The defocus PSF variation with field position will degrade the estimation precision of DFD. Meanwhile, we can see the plots of computed PSFs in Figure 5.6 are fairly consistent with the captured PSFs (solid curves). This verifies our derived Proposition 5.2.2 and confirms that the proposed DFDiff does not rely on the shape of defocus PSFs (Equation 5.9). This property relaxes requirements on the camera lens and enables high precision depth estimation with small, low-end lenses .

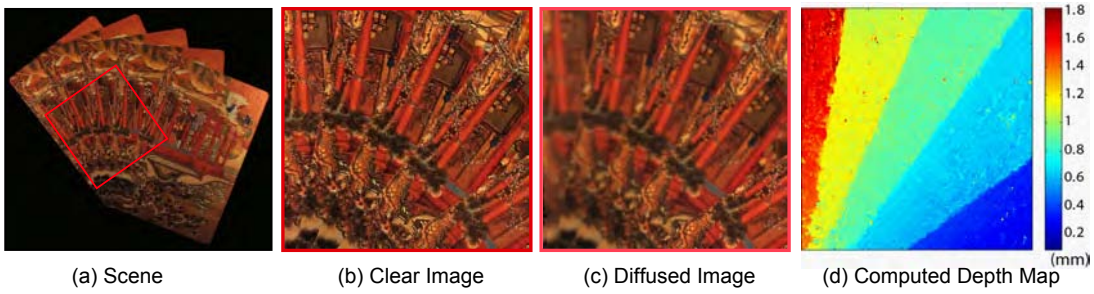


Figure 5.7: Recovered depth map of five playing cards, each of which is $0.29mm$ thick. (a) An overview of the scene. (b) A captured image without a diffuser. (c) A captured image with a 20° Gaussian diffuser. (d) The recovered depth map which has a precision $\leq 0.1mm$

5.5.2 Depth from diffusion: D-SLR Camera

Figure 5.7 shows an example where we use the proposed DFDiff method to estimate the depth map of an artificial scene. Five playing cards are arranged as shown in Figure 5.7(a). Each card is only $0.29mm$ thick. To estimate depths, we captured an image using a Canon EOS 20D D-SLR camera with a Canon EF $50mm$ F/1.8 lens. The distance was set to be $500mm$, which approaches the minimal working range of this camera. The camera was focused at the plane of cards. Note that for this setting, the depth of field is about $6mm$, far larger than the scene depth, and therefore all the cards are in focus. A clear image taken without a diffuser is shown in (b). Then, we placed a 20° Luminit Gaussian diffuser just in front of the first card and captured a diffused image, as shown in Figure 5.7(c). From these two captured images, DFDiff recovers a high-precision depth map, as shown in (d).

According to Equation 5.10, by using the diffuser, we have effectively created a huge virtual lens with F# = 0.12, 15 times larger than the F# of the actual lens. Note that for a regular $50mm$ F/1.8 lens, the depth of field is $6mm$, much larger than the required depth precision. Therefore, DFD cannot be used effectively in this setting.

5.5.3 Depth from diffusion: consumer-level camera

DFDiff imposes fewer restrictions on the camera lens, so that a low-end consumer camera can be used to estimate a high-precision depth map. Figure 5.8 shows a small sculpture of about $4mm$ thickness. For this experiment, we used a Canon G5 camera with a $28.8mm$ F/4.5 lens and a

diffusion of 5° angle. The camera was set up $300mm$ away from the object. The captured focused and diffused images are shown in (b) and (c), respectively. From these two images, we compute the 3D structure of the sculpture of precision $\leq 0.25mm$, as illustrated in (d) and (e). To achieve the same precision in the same scene setting, DFD requires a much larger lens ($F\# \approx 0.5$).

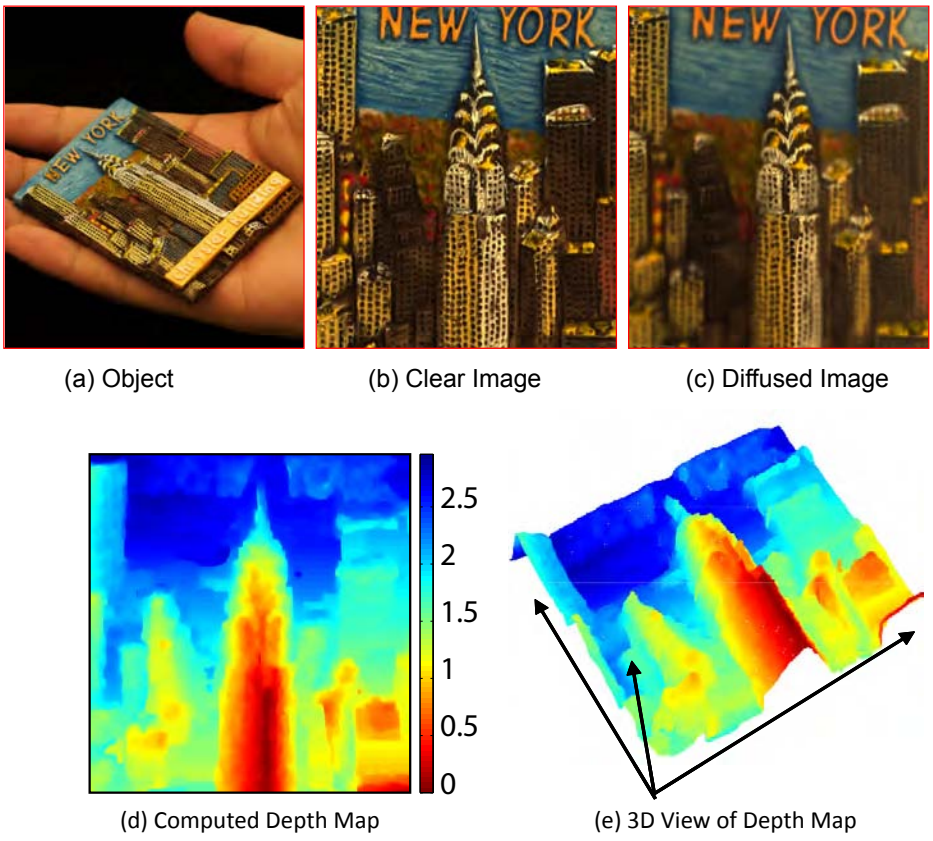


Figure 5.8: DFDiff results for a thin sculpture captured using a Canon G5 camera. (a) Wide view of the sculpture. (b) A clear image without a diffuser. (c) An image captured using a 5° Gaussian diffuser. (d) The computed depth map which has a precision $\leq 0.25mm$. (e) A 3D view of the computed depth map.

5.6 Summary

In this chapter, we have demonstrated that optical diffusers can be used to perform high-precision depth estimation. In contrast to conventional DFD, which either requires a prohibitively large aper-

ture lens or small lens-to-object distances which restricts the FOV, DFDiff relaxes requirements on the camera lens and requires only larger diffusion angles, which are much cheaper to manufacture. Even a low-end consumer camera, when coupled with the proper diffuser, can be used for high-precision depth estimation.

One of the beneficial properties of the DFDiff technique is that depth estimation is measured relative to a proxy object instead of a camera lens, which introduces more flexibility in the acquisition process. However, this same property is also a major drawback since it requires a diffuser to be placed near objects being photographed, which is not possible in many situations.

In our implementation, we have chosen diffusers with Gaussian diffusion functions for simplicity. Diffusers with a variety of diffusion functions are currently commercially available. An interesting question that warrants further investigation is: "What is the optimal diffusion function for depth estimation?". For simplicity, we have used a typical DFD algorithm, which requires two input images. Another interesting topic for further research is how to design diffusers and algorithms that enable depth estimation using only a single image.

Chapter 6

Focal sweep photography for space-time refocusing

6.1 Introduction

Finite DOF of a lens camera leads to defocus blur and often also produces artistic visual experience. It is an effective tool to draw user attention selectively to a specific part of the scene. As a result, it is critical for a photographer to focus at the right depth when images are taken. Many of current displays are interactive in nature. This opens up the possibility for a novel visual representation that allow the users to refocus an image to different depths after capture, so that they can experience the artistic narrow DOF appearance of the scene while simultaneously making available the image detail for the entire image. For this purpose, an image stack of varying focus settings has to be captured or rendered to enable this capacity of interactive image refocusing.

One way to produce focal stack is to capture the entire light field [91] [54]. For example, a plenoptic camera [95] [119] modulates light using lens array in such a way that a 4D light field can be captured by a single 2D image. The captured 4D light field can then be used to render focal stack for image refocusing. But this requires sacrificing the spatial resolution significantly. This is because of the dimensionality gap the captured information (light field) is 4D, while the required information (focal stack) is only 3D. A lot of redundant information is captured by light field cameras.

Our goal is to capture focal stacks directly, with the goal of providing refocusing abilities. We

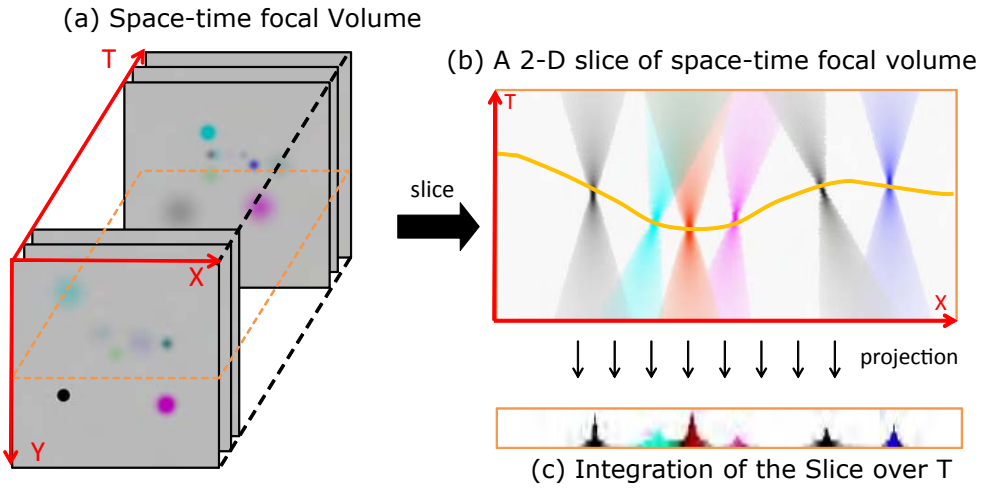


Figure 6.1: Space-time focus volume. (a) A space-time focus volume of a synthetic scene of color balls with motion. Objects move as the focus changes with time in the T dimension. (b) A 2D XT slice of the 3D volume, in which each small ball appears as double-cones. The double-cones of moving balls are tilted. (c) Integrating the volume along the T dimension produces an EDOF image as captured by a typical focal sweep technique. Each object appears sharp in the EDOF image regardless of the depth.

capture a sequence of images during the period of focal sweep. Concatenating all the captured images in the temporal dimension forms a 3Dimensional space-time (*XYT*) volume. Unlike the traditional static focus volume, the space-time focus volume captures object motions as well. Since we are considering dynamic scenes, two effects happen along the third dimension - scene motion, and change of focus. Figure 6.1 (a) shows an *XYT* 3D space-time focus volume of a synthetic scene of colored balls in motion. To peek into the 3D structure, Figure 6.1 (b) shows one 2D *XT* slice of the 3D volume, in which balls appear as double-cones. The apex of a double-cone shows the time when a ball is focused. (The same phenomenon has been observed and well studied in deconvolution microscopy [100]). For objects that are in motion in x direction, these double-cones appear tilted like the cyan cone shown in Figure 6.1 (b). By finding the apexes in the 3D volume (shown as the yellow band in Figure 6.1 (b)), we can obtain a space-time in-focus image. The shape of the band reveals the 3D structure of scene.

Most existing refocusing techniques (e.g., plenoptic cameras [95, 119], camera array [170], depth from defocus [88]) capture images in one moment. The proposed focal sweep camera differentiates itself from these techniques by capturing a space-time focus stack in a longer period when objects can be moving. While capturing an instant of time can be favorable in some situations, capturing a duration of time yields a unique and appealing user experience – the focus transition is so intertwined with object motion that users will see objects move into (or out of) focus when they click to refocus. In addition, our image refocusing is done at sensor resolution without any trade-off in image quality.

Due to object motion and finite capture time, in practice we can only capture or sample a limited number of images from the 3D space-time volume. For this reason, one must decide how many images should be captured, and where the sensor should be positioned at for each capture in order that every object in the depth range will appear focused in at least one of the captured images. In this thesis, we show a capturing and focal sweep strategy that yields an efficient and complete sampling of 3D focus volumes.

Image refocusing displays the right layer from focal stack per user click, where the clicked pixel appears sharp. A major challenge in designing algorithm was to compute an in-focus index map (or depth map) that enables a seamless refocusing experience. First, since users expect that each click will bring them to the image layer where the region boundary is well focused, even in texture-less

regions, there cannot be any holes in the index map. Second, the index precision must be high enough (especially for textured region or region boundary) so that the refocused layer will appear perfectly focused. As we will show in Section 6.6, our algorithm design meets the needs of both these challenges.

We design and build prototypes of focal sweep cameras, whose focus can be swept at a proper speed with synchronization to image capturing. A collection of focal sweep photographs has been captured by using our prototypes, and users can refocus these images interactively on *www.focalsweep.com*.

6.2 Related work: Focal sweep and focal stack

A conventional lens camera has a finite depth of field. A variety of EDOF techniques have been proposed to extend depth of field in the past several decades [27, 36, 41, 48, 60, 75, 103, 106, 127]. Focal sweep is one of the typical EDOF techniques. A focal sweep EDOF camera captures a single image when its focus is quickly swept over a large range of depth. Hausler [70] extended DOF of microscope by sweeping the specimen along optical axis during exposure. Nagahara et al. [106] extended DOF for consumer photography by sweeping the image sensor. Nagahara et al. [106] also show that the point-spread-function (PSF) of a focal sweep camera is depth invariant, allowing one to deconvolve a captured image with a single PSF to recover a sharp image without knowing the 3D structure of the scenes. We build a similar imaging system as in [106], but use it to capture image stacks.

Several techniques have been proposed to capture a stack of images instead of a single EDOF image for extended depth of field and 3D reconstruction. In deconvolution microscopy, for example, a stack of images of specimens are captured at different focus settings to form a 3D image [100, 150]. 3D point-spread-functions in the 3D images are shown to be depth invariant double-cones. By deconvolving with the 3D PSF, a sharp 3D image can be recovered. We observe a similar double-cone structure in image stacks captured using our imaging system. To produce an all-in-focus image from a focal stack, Kuthirummal et al. [82] first average all images in a focal stack to produce a single EDOF image as captured by an EDOF focal sweep camera and then recover an all-in-focus images by deconvolution. Guichard et al. [60] and Cossairt and Nayar [35] make use of chromatic aberration to capture images of different foci in the three color channels with a single shot, and then

by combining the sharpness from all color channels to produce an all-in-focus images. Agarwala et al. [5] propose using a global maximum contrast image objective to merge a focal stack into a single all-in-focus images.

Hasinoff et al. [69] compare the optimality of various capture strategies for reducing optical blur in a comprehensive framework where both sensor noise model and deblurring error are taken into account. Their analysis and future analysis in [83] show that focal stack photography has two performance advantages in extending depth of field over one-shot photography: 1) it allows one to capture a given DOF faster; 2) it achieves higher signal-to-noise ratio (SNR) in a given exposure time.

Hasinoff and Kutulakos [68] consider the problem of minimizing the time to capture a scene with a given DOF and a given exposure level. This is highly related to the optimization problem in our technique and similar analysis on camera DOF can be seen in both papers. While Hasinoff and Kutulakos [68] emphasize on the lens f-number and the number of images for capturing focal stack, we optimize the speed of focal sweep in synchronization with image exposure, for a given lens f-number.

Kutulakos and Hasinoff [83] use a similar algorithm as in [5] to synthesize EDOF images from focal stacks by assuming that scenes are static. While their algorithms are optimized to produce artifact-free images with minimal blur, our algorithm is proposed to yield a seamless space-time refocusing experience.

Computing the focus measure is a critical technique for all depth from focus approaches. There are several difficulties associated with it. The space-scale effect [125] presents a major difficulty since it can lead to depth ambiguities at different scales. In addition, image patches at depth discontinuities may cross multiple depth layers and make focus measure inaccurate. Furthermore, since focus measure does not reveal anything about depth in non-textured regions, techniques such as plane fitting [162], graph-cut [22], and belief prorogation [177]) have been employed to fill the resulting holes in the depth map.

6.3 Space-time focus volume, focus sampling, and refocusing

Consider concatenating all the images captured during focal sweep along the temporal dimension. This forms a 3D space-time volume that encodes more visual information about scenes than a single image. Figure 6.1 (a) shows a space-time focus volume of a synthetic scene of colored balls in motion and (b) shows one 2D *XT* slice of the 3D volume, in which balls appear as double-cones. As mentioned in the introduction, the double-cones appear tilted for objects in motion in both x and y directions; by finding the apexes in the 3D volume (shown as a yellow band in Figure 6.1 (b)), we can obtain a space-time in-focus image; and the shape of the band reveals the 3D structure of the scene.

6.3.1 Space-time focal stack and focus sampling

Within a given time budget, one can only capture a finite number of images during focal sweep. This is because of the limited framerate of the sensor, and also SNR considerations. The photographer must decide how many images to capture and where the sensor should be positioned at for each capture point so that every object appears focused in at least one of the captured images. This is, in essence, a sampling problem of the 3D focus volume. We argue that an ideal capture should satisfy two conditions:

- Completeness: the DOFs of all captured images should sum up to cover the entire desired depth range. If the desired depth range is DOF^*, we have

$$DOF_1 \bigcup DOF_2 \bigcup DOF_3 ... \bigcup DOF_n \supset DOF^*, \tag{6.1}$$

  where $\bigcup$ denotes a union operation.

- Efficiency: No two DOFs should overlap, so only a minimal number of images are required.

$$DOF_1 \bigcap DOF_2 \bigcap DOF_3 ... \bigcap DOF_n = \varnothing, \tag{6.2}$$

  where $\bigcap$ denotes intersection.

Hasinoff and Kutulakos [68] refer to an image sequence as *Sequence with Sequential DOFs*, if the end-point of one image's DOF is the start-point of the next image's DOF. The ideal capture

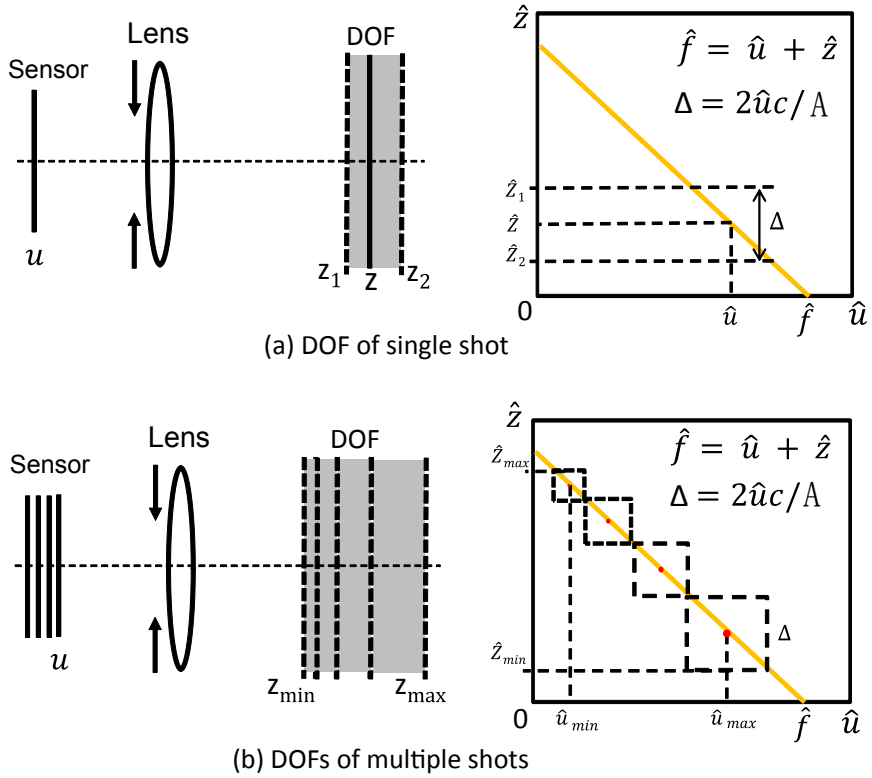


Figure 6.2: Efficient and complete focus sampling. (a) Left: A geometrical illustration of depth of field. Objects in the range $[Z_1, Z_2]$ will appear focused when u and z satisfy the Thin Lens Law. Right: The Thin Lens Law is shown as an orange line in the reciprocal domain. Z_1 and Z_2 can be easily located in the reciprocal domain (or in diopter) by $|\hat{Z}_i - \hat{Z}| = 2\hat{u}c/A$. (b) In order to have an efficient and complete focus sampling, the DOFs of consecutive sensor positions (e.g., $\hat{v}_{i-1}, \hat{v}_i, \hat{v}_{i+1}$) must have no gap or overlap.

sequence described above is a sequence with sequential DOFs that covers the entire desired depth range.

We start our DOF analysis from the Thin Lens Law, $1/f = 1/u + 1/z$, where f is the focal length of the lens, u is the sensor-lens distance, and z is the object distance. As is a common practice, we transform the equation to the reciprocal domain $\hat{f} = \hat{u} + \hat{z}$, where $\hat{x} = 1/x$. In the reciprocal form, the Thin Lens Law is a linear equation. The reciprocal of object distance, $\hat{z}$, is often expressed in the unit of diopter $(1/m)$. Depth of field, $[z_1, z_2]$, is the depth range where the blur radius is less than the circle of confusion, c. In this thesis, we used the pixel size as the diameter of the circle of confusion (common practice in imaging). For a given sensor-lens distance $\hat{u}$, the DOF in the reciprocal domain, $\hat{z}_1$ and $\hat{z}_2$, can be derived as:

$$\hat{z}_1 = \hat{z} + \hat{u} \cdot c/A \tag{6.3}$$

$$\hat{z}_2 = \hat{z} - \hat{u} \cdot c/A \tag{6.4}$$

where A is the aperture diameter of the lens. Both the position and range of DOF changes with the sensor position. Figure 6.2 (a) shows the geometry of DOF for sensor positions u on the left and illustrates the DOF in the reciprocal domain on the right. The yellow line in the figure represents the Thin Lens Law. According to Eqn 6.3, for an arbitrary sensor position $\hat{u}$, the size of DOF in the reciprocal domain is $\Delta = 2 \cdot \hat{u} \cdot c/A$.

For an efficient and complete focus sampling, we require that each pair of consecutive DOFs have no overlap and no gap as shown in Figure 6.2 (b). From Eqn 6.3, we derive:

$$|\hat{u}_i - \hat{u}_{i+1}| = |(\hat{f} - \hat{z}_i) - (\hat{f} - \hat{z}_{i+1})| \tag{6.5}$$

$$|\hat{u}_i - \hat{u}_{i+1}| = |\hat{z}_i - \hat{z}_{i+1}| \tag{6.6}$$

$$|\hat{u}_i - \hat{u}_{i+1}| = (\hat{u}_i + \hat{u}_{i+1}) \cdot c/A, \tag{6.7}$$

where $\hat{u}_i$ and $\hat{u}_{i+1}$ are the focus centers of two consecutive DOFs.

In consumer photography, we have $z \gg u$ and so $\hat{u}_i \approx \hat{f}$. By approximating Eqn 6.7 we have:

$$\hat{u}_i \cdot \hat{u}_{i+1} \cdot |\hat{u}_i - \hat{u}_{i+1}| = \hat{u}_i \cdot \hat{u}_{i+1} \cdot (\hat{u}_i + \hat{u}_{i+1}) \cdot c/A \tag{6.8}$$

$$|u_{i+1} - u_i| = (u_i + u_{i+1}) \cdot c/A \tag{6.9}$$

$$\delta u \approx 2 \cdot f \cdot c/A \tag{6.10}$$

$$\delta u \approx 2 \cdot c \cdot N, \tag{6.11}$$

where $N = f/A$ is the f-number of the lens. Equation 6.11 shows that to achieve an efficient and complete focus sample, we need to move the sensor by a constant amount between successive image captures. The step size is determined by the pixel size and f-number. Notice that this is a constant step in the normal domain. In the reciprocal domain, the step is not constant as shown in Figure 6.2 (b). For a fixed framerate P, this indicates that a sensor should be swept at a constant speed:

$$s = \frac{\delta u}{\delta t} = 2 \cdot c \cdot N \cdot P \tag{6.12}$$

.

If the time-budget is too small (or the sensor is moving too slowly) to perform a complete focus sweep, deblurring must be done to recover the sharpness of objects in DOF gaps. Hasinoff et al. [69] proposed a comprehensive framework for optimizing focus sampling in this case by considering the noise model, capturing overhead, and the effect of deblurring. In this chapter, we concentrate on the problem of how to sample the focal volume in an efficient and complete manner for space-time image refocusing. By avoiding aggressive deblurring in the process, we can save a large amount of computation and produce higher quality refocusing results that are more natural and artifact-free.

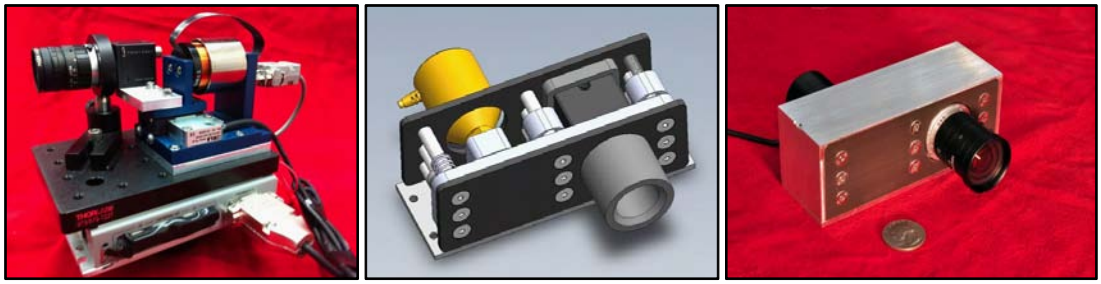
6.3.2 Space-time in-focus index map and refocusing

In a system where the speed of focal sweep is much faster than the speed of an object's motion along the optical axis (z), the object appears in focus only once in the focus volume (or, equivalently, in only one image of the focal stack). Let $F_1, F_2, \ldots, F_k$ be k images in the focal stack. For each pixel, we find the index of the frame where the pixel is best focused. We call this the in-focus index of the pixel. The space-time in-focus index map is then defined as the in-focus indices for all the pixels.

In a dynamic scene, both the focus and objects themselves are free to move, which leads to ambiguities in the definition of the space-time in-focus index map. Hence, there are different ways of defining the space-time in-focus index map. Here we list three of the possible definitions:

1. For each pixel (x, y), look into a small tube at (x, y) in the 3D *XYT* focal stack and find the layer T that appears the sharpest. Then, $T(x, y)$ is the in-focus index map for the focal sweep.

2. Explicitly consider object motion. At an arbitrary layer (or time) t, for any object at a spatial location (x, y), we could track this object and find that the object is best focused at the layer



(a) A side view of Prototype 1 (b) CAD rendering of Prototype 2 (c) A side view of Prototype 2

Figure 6.3: Two focal sweep camera prototypes. (a) Prototype 1 drives sensor sweep using a voice coil; (b) Prototype 2 drives lens sweep using a linear actuator.

(or time) t' and spatial location (x', y'). In this case, the in-focus index map is $t' = T_1(x, y, t)$, which is a 3D index map.

3. Since multiple objects can be observed at a location (x, y), one can track all these objects to the layers where they are best focused. Then, among all the final layers, pick the layer closest to the present layer. In this case, we have the space-time in-focus index map as $t' = T_2(x, y, t) = \arg\min_t |T_1(x, y, t) - t_0|$.

In this thesis, we choose the first definition for simplicity. A refocusing viewer takes a user click (x, y) as input, finds the next frame t by $T(x, y)$, and smoothly transitions the image from the current frame to the next frame. The other two choices of definition can yield different refocusing behaviors and user experiences. Ideally, the choice should be made based on user intention or preference when they click on a pixel in the viewer. We leave the study of the other two (or more) definitions and their impacts on user experience to future work.

With this definition, a space-time in-focus index map $T(x, y)$ is a mapping from a spatial location (x, y) to a temporal point $t = T(x, y)$. But notice that since the focal volume is captured in a duaration, objects can be moving as the focus plane sweeps. As a result, users will be able to observe objects move with refocus variation.

6.4 Focal sweep camera

6.4.1 Prototypes

Focal sweep can be implemented in multiple ways. One way is to directly sweep the image sensor. A variety of actuators such as voice coil motors, piezoelectic motors, ultrasonic transducers, and DC motors could be used to translate the sensor in a designed manner during capture duration. Another way is to sweep camera lens. With the auto-focus mechanism that is commonly built in many commercial lenses, it may also be programmed to perform focal sweep photography. Liquid lenses Ren and Wu [134], Ren et al. [135] are yet another way of performing focal sweep, and they are power efficient. Liquid lenses focus at different distances when different voltages are applied to them.

We built two prototype focal sweep cameras as shown in Figure 6.3. Prototype 1, as shown in Figure 6.3 (a), uses a Fujinon HF9HA-1B, 9mm, F/1.4, c-Mount lens, and a Pointgrey Flea 3 camera with a max resolution of 1328×1048. Its sensor is driven by a voice coil actuator (BEI LA15-16-024). This setting is similar to the one used in [106] for capturing extended depth of field. The sensor is tethered to a laptop via a USB 3.0 cable and synced with the motor start/stop signal. The voice coil motor and the motor controller are able to translate the sensor at the speed of 1.47 mm/s. In almost all scenes that we have experimented, the sensor motion is less than 0.3mm, which can be completely in 0.21 second. The major advantage of this implementation is that all of the parts are off-the-shelf components. This first prototype demonstrates that a focal sweep camera can be built with minimal effort. A collection of focal sweep photographs captured by this prototype are shown on the website *www.focalsweep.com*.

Prototype 2, as shown in Figure 6.3 (b), is a more compact design, in which a sensor secured on a structure and the lens can be translated during the sensor's integration time. In this prototype we use a compact linear actuator instead of a voice coil motor, allowing us to reduce the camera's overall size. The same lens and camera as in Prototype 1 are used. During the integration time, the sensor is translated from the near focus position to the far focus position. With this prototype, we are able to translate the sensor at a top speed of $0.9mm/s$. The major advantage of this implementation is its compactness and its close resemblance to existing camera architectures.

6.4.2 Camera settings

As in conventional photography, users first determine the frame rate P and f-number N according to the speed of object motion, the lighting condition, and the desired amount of defocus in the captured images for each scene. Then, the ideal speed of sensor sweep s can be computed using Equation 6.12. Note that s is independent of camera focus and distance range of scenes. This independence makes configuration friendly to users.

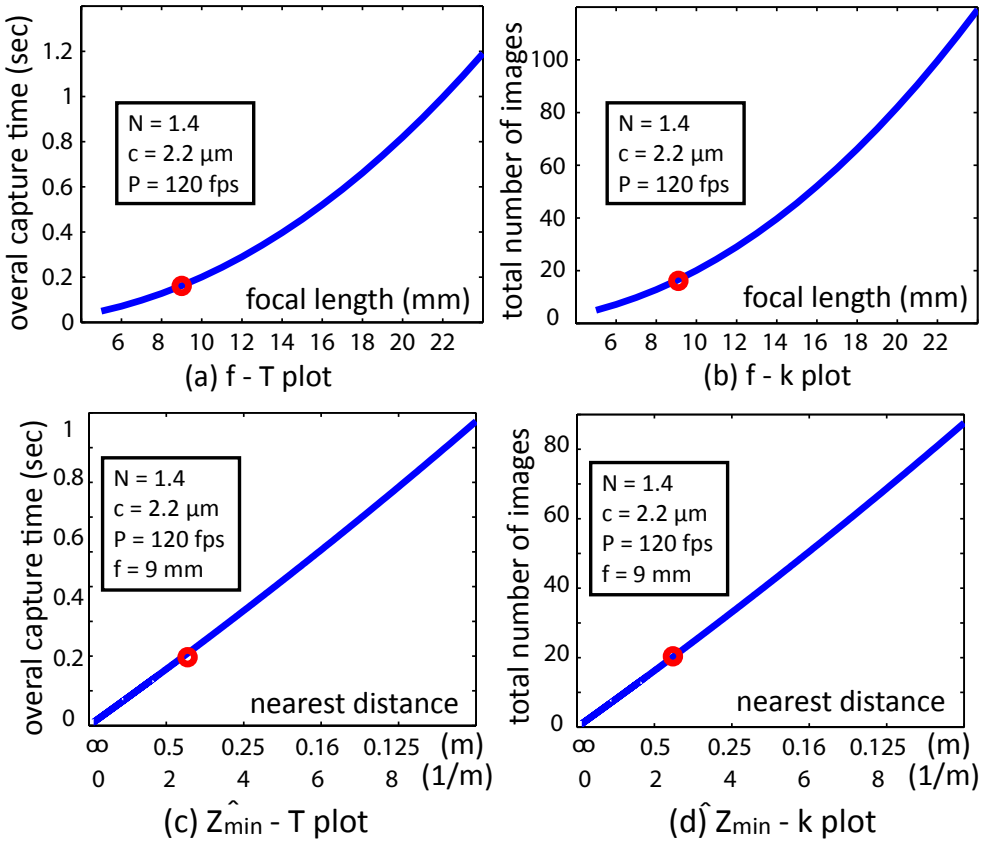
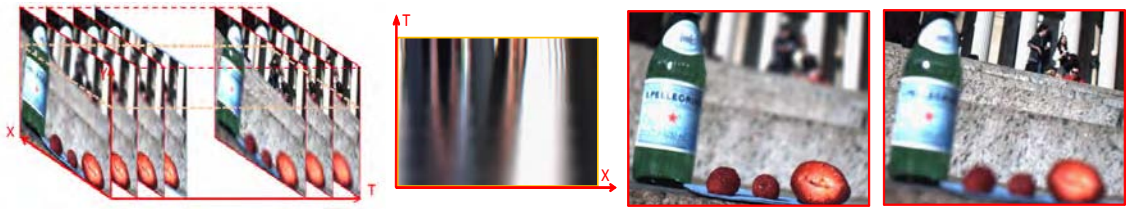


Figure 6.4: For a given pixel size, frame rate, and f-number, the overall capture time and total image count are highly related to focal length and scene distance range. (a) shows the $f - T$ plot of the overall capture time T with respect to focal length f to cover a wide depth range from $0.4m$ to infinity. (b) shows the $f - k$ plot of the total image number k with respect to focal length f to cover a wide depth range from $0.4m$ to infinity. (c) and (d) show the plots of overall time T and total image number k with respect to the depth range (in both diopter and meter), respectively (f = 9mm). In each plot, the red spot indicates the most typical setting in our implementation.



(a) A Space-time focal stack (b) 2D slice of the stack (c) The first frame (d) The last frame

Figure 6.5: A sample space-time focal stack captured using our focal sweep camera prototype 1. (a) A space-time focal stack of 25 images; (b) A 2D slice of the 3D stack; (c) The first frame of the stack where the foreground is in focus; (d) The last frame of the stack where the background is in focus. The capturing frame rate is $120fps$. It took the focal sweep camera about $0.2sec$ to capture the whole sequence.

Although the sweep speed is independent of camera focus and scene depth range, the number of images, k, and the overall time to capture the image stack, T, are highly related to the camera focus setting and the scene depth range. Consider a depth range from 0.4m to infinity, Figure 6.4 (a) shows how the overall capture time T varies with camera focal length f in a camera where $N = 1.4$, $c = 2.2\mu m$, and $P = 120$ fps. T increases proportionally to the square of f. Figure 6.4 (b) plots the total number of captured images, k, with respect to focal length f for the same camera. Again, k is linearly proportional to f^2.

In the most common scenarios, the desired scene ranges from a certain distance, Z_{min}, to infinity. Figure 6.4 (c) and (d) plot the capture time T and total image count k with respect to $\hat{Z}_{min}$, which is the inverse of distance (or dioper). We can see that both are linear. The closer the foreground is to the camera the longer the capture time. The x-axis is labeled in the unit of both dioper $(1/m)$ and distance (m) for easy reference.

It can be noted from the figure that the required capture time and image counts have a huge range at different settings. The red dot in each plot indicates a typical setting in our implementation. We use a $9mm$ lens and our scenes' depths range from $0.4m$ to infinity, so it takes us $0.2sec$ to capture 20 images. Our prototypes are also able to capture scenes with smaller Z_{min}, but it takes a long time to capture the sweep as shown in Figure 6.4 (c). Figure 6.5 illustrates an sample of space-time focal stack that was captured using our first prototype camera.

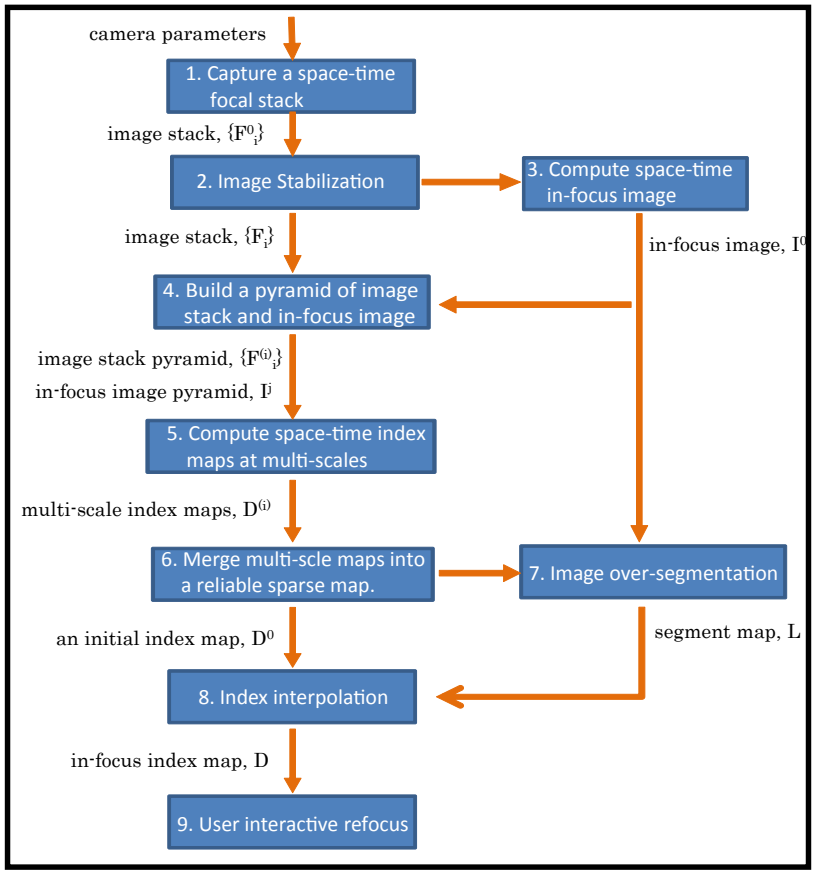


Figure 6.6: A diagram illustrating the process from capturing a space-time focal stack, to generating an in-focus index map, and to interactive image refocusing.

6.5 Algorithm

Figure 6.6 shows an overview of the newly proposed algorithm. After a stack of images, $\{F_i^0\}$, are captured, we first apply a typical multi-scale optical flow algorithm to estimate frame-to-frame global transformations to account for hand-shake, then stabilize the image stacks. The stabilized image stack $\{F_i\}$ is then used to compute a space-time in-focus image (Section 6.5.1) and index maps at various scales (Section 6.5.2). In Section 6.5.3, we describe a new approach to merging multi-scale index maps into one high-quality index map.

There are two key ideas in the algorithm. First, we use a pyramid strategy to handle non-textured regions. For each pixel, we estimate its index (the frame it is best focused) at multiple scales. Due to the space-scale effect [125], the index may not be consistent at different scales, especially in

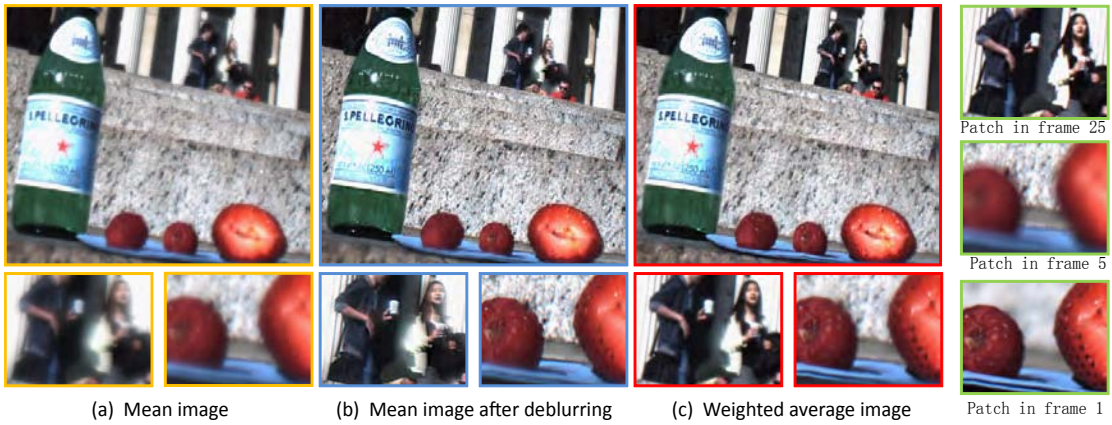


Figure 6.7: Space-time in-focus images computed using different approaches and their close-ups. (a) The mean of all images in the stack; (b) The mean image deconvolved using an integral PSF; (c) Weighted average of all images in the stack; (d) The best focused patches in the captured focal stack.

regions with weak textures, depth discontinuities, or object motions. This inconsistency is one of the fundamental difficulties in the algorithm's design, and we show a simple yet effective solution.

Second, at any given scale, we propose a novel approach to computing the index map. In literature, it is common to first estimate an index map (or depth map) using focus measure, and then use the index map to produce an all-in-focus image [5, 68]. In this thesis, however, we take a different strategy. We first compute an all-in-focus image without knowing the index map, and then use the all-in-focus image to estimate the index map. We will show the advantage of using this new strategy.

6.5.1 Space-time in-focus image

Given a focal stack, we first compute a space-time in-focus image without the index map. The idea is inspired by the EDOF technique using focal sweep. Kuthirummal et al. [82] show that the mean of a focal stack preserves image details, and deconvolve the averaged image with a $(1/x)$-shape integral point-spread-function (IPSF) to recover an all-in-focus EDOF image without knowing the depth map. This approach is further shown to be robust in regions of depth edges, occlusions, and even object motion. In Figure 6.7, we show the mean image of a space-time focal stack (a) and the EDOF image after deconvolution (b).

Although both (a) and (b) preserve most high frequency information, the average image yields a low contrast (especially when the number of images increases), and the deconvolved EDOF image (b) is prone to image artifacts. In addition, deconvolution is computationally expensive, especially for mobile devices. In this thesis, we compute a space-time in-focus image as a weighted sum of all images:

$$I(x) = \frac{\Sigma_i W_i(x) \cdot F_i(x)}{\Sigma_i W_i(x) + \epsilon}, \tag{6.13}$$

where the weights $W_i(x)$ are defined as the variance of the Laplacian patch $\triangle(\mathbf{P}_i(x, d))$:

$$W_i(x) = \mathcal{V}(\triangle \mathbf{P}_i(x, d)). \tag{6.14}$$

$\mathbf{P}_i(x, d)$ here represents a patch of size d centered at x in the i^{th} frame. With this strategy, severely blurred patches will carry much less weight than sharper patches do, reducing the hazy effects that one can see in the average image from Figure 6.7(a). As shown in (c), the weighted sum is sharp and has high contrast even without deconvolution. Although the weighted sum (c) is sometimes not as sharp as the deblurred image (b), it avoids the risk of deconvolution artifacts and reduces the halo effects introduced by object motion. It is important to note that our final goal is not to produce an all-in-focus image, but using an all-in-focus image to compute the in-focus index map. A decent all-in-focus image free of high-frequency artifacts is essential for this purpose.

6.5.2 Space-time in-focus index maps at various scales

We use the computed all-in-focus image $I(x, y)$ to help estimate in-focus index map. For each pixel (x, y), we look for the frame where its surrounding patch is most similar in high frequencies to that in $I(x, y)$. Then, the in-focus index map $M(x)$ is estimated as:

$$M(x, y) = \arg\min_i S(F_i(x, y), I(x, y)), \tag{6.15}$$

where S measures the high frequency similarity between F_i and I at each pixel and is defined as

$$S(P, Q) = |\triangle(\mathbf{P} - \mathbf{Q})| \otimes \sqcap(r), \tag{6.16}$$

where $\mathbf{P}$ and $\mathbf{Q}$ denote the patches at P and Q, respectively, $\otimes$ is convolution, and $\sqcap(r)$ is a pillbox function of radius r. The key idea here is to measure the similarity in high frequencies. The weighted mean image preserves good high frequencies as in the best focused layer even at depth

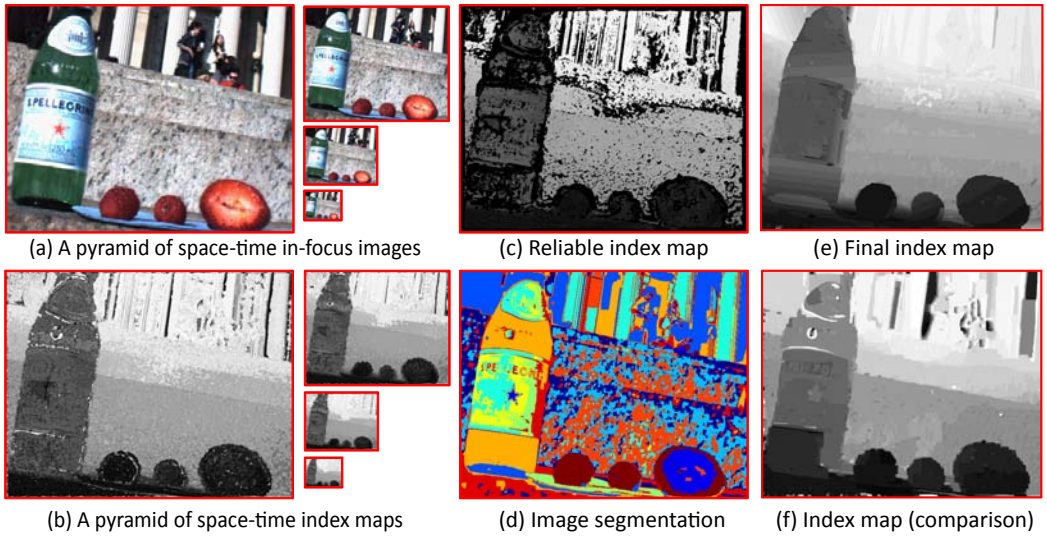


Figure 6.8: (a) A pyramid of space-time in-focus images; (b) A pyramid of space-time index maps; (c) A reliable index map that is computed from (b) using index consistence; (d) An over-segmentation of the full-resolution in-focus image; (e) Our final depth map computed from (c) and (d) by hole-filling; (f) An index map computed using a traditional algorithm which uses difference-of-Gaussians as focus measure.

discontinuities and for moving objects. By convolving with $\sqcap(r)$, we consider a neighborhood in processing each pixel.

6.5.3 Merging and interpolating index maps

Due to the space-scale effects and depth discontinuity, the index map computed at different scales (or different neighborhood size r) can be significantly different. Figure 6.8 (b) shows the index map pyramid $M^{(i)}, i = 1, 2, \ldots, k$, where k is the total level of the pyramid. At each level, the focal stack reduces the spatial resolution by 2×2 from its upper level. The index maps at different scales are significantly different, especially at depth boundaries. It is a challenging problem to pick a right scale for each pixel.

We propose a novel multi-scale technique to solve this problem. First, we construct a reliable but sparse index map D^0 by only accepting indices that are consistent in all levels:

$$D^0(x) = \begin{cases} \overline{M_i(x)}, & \text{if } \max[M_i(x)] - \min[M_i(x)] < \tau \\ \\ \emptyset, & \text{otherwise} \end{cases} \tag{6.17}$$

τ is set as a small number to enforce consistence. One sample is shown in Figure 6.8 (c). (We use $d = 7, k = 7, r = 5$ in our implementation.) The pixels with no index assigned are shown in black. The observation is that the index map is dense in textured region, and sparse in non-textured regions and depth boundaries. Second, we over-segment the in-focus image $I(x, y)$. Third, in each segment, we fill the holes in D^0 by interpolation according to two simple rules:

- If the segment has at least m valid (and reliable) indices, do interpolation by fitting a plane to the valid indices.
- If the number of valid indices is less than m, do nearest neighbor interpolation.

This gives us the final index map $D(x, y)$. Our observation is that segments in a textured region have many reliable indices in D^0, which yield a reliable plane fitting; segments in a texture-less region or depth discontinuities have few indices and so the hole-filling process proprogates index information from the region boundary.

By doing nearest neighbor interpolation, we avoid smoothing out the index map in these regions. It is important to note that a smoothed index map at depth boundary must be avoided, because it would lead focus to a middle point where neither foreground nor background is well focused. Nearest neighbor interpolation may not be able to produce an accurate spatial boundary between foreground and background, but fortunately, users are much more tolerant to this spatial inaccuracy. This is because the precision of user input itself (e.g., finger tapping on a touch screen) is usually much lower than image resolution.

Figure 6.8 (d) shows a result of image over-segmentation using Graph-cut (d), and Figure 6.8 (e) shows the index map after interpolation. We can see that the index map is sharp at depth boundary, and smooth in non-textured regions.

With the estimated index map $D(x, y)$, we can do image refocusing. In the refocusing viewer, for any pixel (x, y) that a user clicks, we transition the displayed image from the present image to the image indexed by $D(x, y)$. The transition is made smooth by sequentially displaying the images between the present index to $D(x, y)$. We have made our refocusing viewer available online at *www.focalsweep.com*.

6.6 Experiments

In all the experiments shown in this chapter, we set $m = 10, \tau = 1, d = 7, r = 5$. We compare index maps computed using the proposed technique with that using a traditional depth estimation algorithm, which maximize a simple focus measure. There are various definitions of focus measures [115, 116, 157, 175]. We adopted the one used in the photomontage method [5], which defines focus measure as a simple local contrast according to the Difference-of-Gaussians filter, and we further polished the results using Graph-cut [22]. Graph-cut as a global optimization technique helps to fill up the holes and smooth the index map, shown in Figure 6.8 (f). Our results (e) show better results in non-textured or specular regions and depth discontinuities, which are important for image refocusing.

Figure 6.9 shows more space-time focal stacks that we captured using Prototype 1, as well as the computed in-focus images and index maps. In the index maps that the proposed algorithm computes, there are no obvious holes or artifacts, even in textureless regions. The index map is also sharper

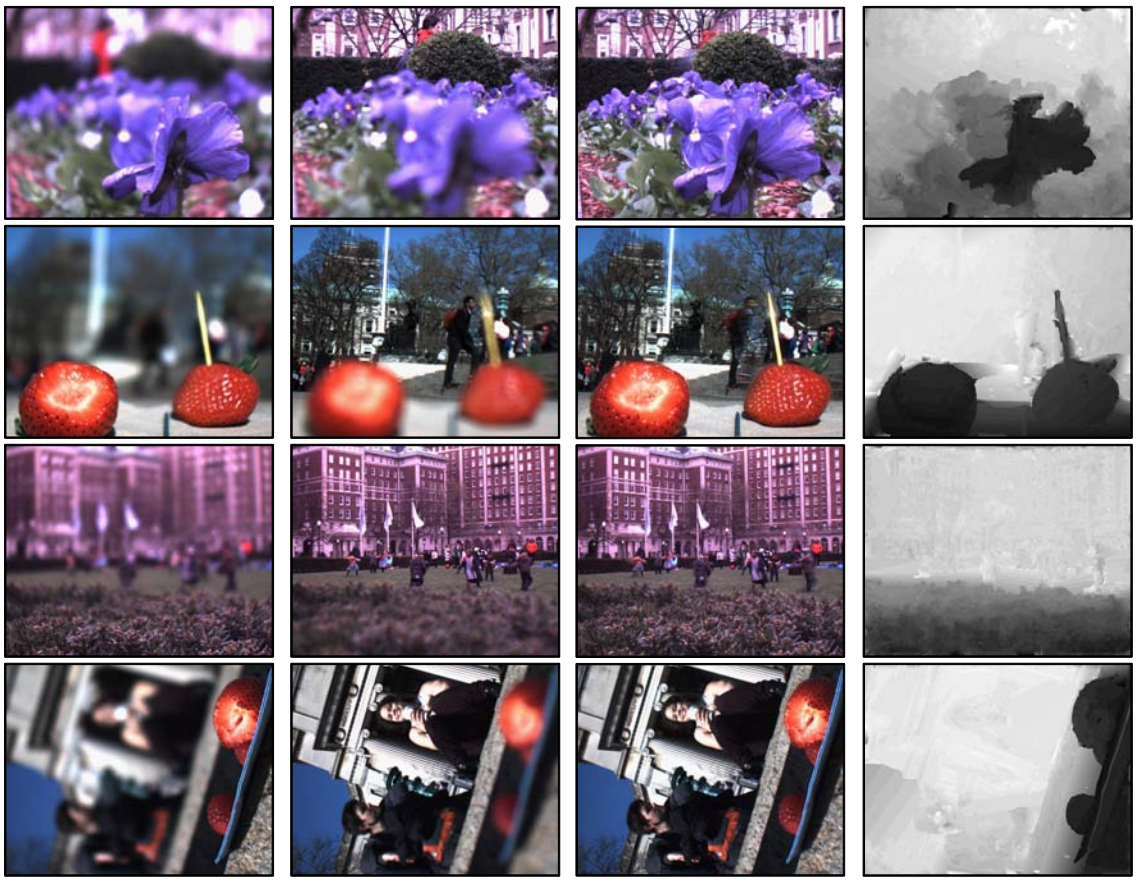


Figure 6.9: More experimental results. Each row corresponds to a scene. From left to right, (a) and (b) are the first and last frames captured with focal sweep, (c) are the computed space-time in-focus images, and (d) are the estimated space-time in-focus index maps. The resulting index maps are used for image refocusing, as demonstrated on our website *www.focalsweep.com*.

at depth boundaries. In this chapter, we only show the computed index maps and the all-in-focus images. For interactive viewing of the focal stacks, please visit the website *www.focalsweep.com*.

6.7 Summary

We present a focal sweep imaging system to capture space-time focal stacks for image refocusing. The proposed camera sweeps focus at a sufficiently high speed so that the summed DOF of the captured focal stack efficiently covers the entire desired depth range. The major benefit of this design lies in the fact that the camera directly captures all the images that are required for image refocusing. By avoiding the dimension gap in capturing and image synthesis in image processing (which are common in many other competing designs), this design provides users high-quality full-resolution images at every focus with minimal computation cost.

Due to object motion, each pixel that a user clicks on might correspond to different objects at different focus layers (or time points). For example, a defocused object in motion often appears blended with its background object, and there would be cases when it is preferred to estimate object motion and do image refocusing along the estimated motion trajectory. Solving the ambiguity often requires a deeper understanding to user intentions. In this thesis, however, we choose to stay simple by not explicitly considering object motion in the algorithm design. There are other possible refocusing choices as discussed in Section 6.3.2, which deal with the ambiguities in different manners. We decide to leave them as future work.

A more compact and promising implementation could be made by making use of the auto-focus mechanism, which commonly exists in many commercial lenses. To do so, one will need the capability to control and synchronize focus with image capturing.

Chapter 7

Conclusions

A computational camera combines novel optics and computation to encode more useful visual information in the captured images. The design space for the optics of computational cameras is so large and the formulation of image formation becomes so complicated that the design and optimization of computational cameras remains part science and part art.

Despite the complexity of computational camera designs, point spread function (PSF) yields an efficient and simple way to characterize these imaging systems. I therefore propose optimizing computational camera designs for scene recovery through point spread function (PSF) engineering. To make this case, I first addressed two keys questions in this thesis:

- What are good PSFs for image recovery? In answering this question, I derived a close-form PSF evaluation criterion for image deblurring. This criterion is comprehensive and accounts for the effects of image deblurring and image noise as well as natural image statistics. In line with this criterion, I have been able to optimize the pattern of lens aperture for defocus deblurring. Experiments show that we can recover more texture details from defocused images by using the optimized coded apertures.

- What are good PSFs for depth recovery? In answering this question, I derived a close-form PSF evaluation criterion for depth from defocus (DFD). I then use this criterion to solve for an optimized pair of coded apertures for DFD. Via simulations and experiments I demonstrate that a camera with an optimized coded aperture pair is able to not only produce depth maps of significantly greater accuracy and robustness, but it also produces high-quality all-focused

images.

Through the PSF optimization, I have significantly improved the performance of defocus deblurring and depth from defocus. I then further proposed using an optical diffuser to modulate the PSFs to overcome a fundamental limit of DFD in depth precision. This led to a novel depth recovery technique – referred to as depth from diffusion (DFDiff). Compared to the traditional DFD, DFDiff is able to recover the depth at a high precision without a large lens, and is insensitive to lens aberration. One drawback of DFDiff is that it requires the flexibility to place a diffuser in the scene.

The finite depth of field (DOF) of a lens camera leads to defocus blur, but this effect also produces an artistic visual experience. In this thesis, I proposed a focal sweep camera design for space-time image refocusing. This technique allows users to experience the artistic narrow DOF appearance of the scene while simultaneously making available the image detail for the entire image.

Appendix A

A Proof of Evaluation Criterion for Defocus Deblurring (Equation 3.15)

Since ζ is a matrix of Gaussian white noise, we evaluate the quality of recovery using the expectation of the L_2 reconstruction error with respect to the random matrix ζ:

$$R(K, F_0, C) = \mathop{\mathbb{E}}_{\zeta}[\|\hat{F}_0 - F_0\|^2], \tag{A.1}$$

where $\mathbb{E}$ denotes expectation. Substitute $\hat{F}_0$ using Equation **??**, we obtain

$$R(K, F_0, C) = \mathop{\mathbb{E}}_{\zeta}\left[\left\|\frac{\zeta \cdot \bar{K} - F_0 \cdot |C|^2}{|K|^2 + |C|^2}\right\|^2\right], \tag{A.2}$$

When ζ is assumed to be a Gaussian white noise $N(0, \sigma^2)$, we have

$$R(K, F_0, C) = \left\|\frac{\sigma \cdot \bar{K}}{|K|^2 + |C|^2}\right\|^2 + \left\|\frac{F_0 \cdot |C|^2}{|K|^2 + |C|^2}\right\|^2. \tag{A.3}$$

Since F_0 is sampled from the image space, we are actually looking for a C to minimize the expectation of R with respect to the image distribution:

$$R(K, C) = \mathop{\mathbb{E}}_{F_0}[R(K, F_0, C)] = \int_{F_0} R(K, F_0, C) d\mu(F_0), \tag{A.4}$$

where $\mu(F_0)$ is the measure of the sample F_0 in the image space. According to the $1/f$ law of natural images [139][154], we know that the expectation of $|F_0|^2$, $A(\xi) = \int |F_0(\xi)|^2 d\mu(F_0)$ exists we have

$$R(K, C) = \left\|\frac{\sigma \cdot \bar{K}}{|K|^2 + |C|^2}\right\|^2 + \left\|\frac{A^{1/2} \cdot |C|^2}{|K|^2 + |C|^2}\right\|^2. \tag{A.5}$$

Substitute $|C|^2$ by σ^2/A and rearrange the equation, we will get

$$R_\xi(K,\sigma) = \Sigma_\xi \frac{\sigma^2}{|K_\xi|^2 + \sigma^2/A_\xi}, \tag{A.6}$$

where ξ is the frequency.

Appendix B

A Proof of Aperture Evaluation Criterion for Depth from Defocus (Equation 4.5)

Given a coded aperture pair (K_1, K_2), a ground truth blur size d^*, and a noise level σ, the energy E corresponding to a hypothesized blur estimate d is as follows:

$$\begin{aligned} & E(d|K_1^{d^*}, K_2^{d^*}, \sigma) \\ =& \sum_{\xi} \frac{A \cdot |K_1^d \cdot K_2^{d^*} - K_2^d \cdot K_1^{d^*}|^2}{\sum_i |K_i^d|^2 + C} \\ +& \sum_{\xi} \frac{\sigma^2 \cdot (\sum_i |K_i^{d^*}|^2 + C)}{\sum_i |K_i^d|^2 + C} + n \cdot \sigma^2. \end{aligned}$$

Proof:

$$E(d|K_1^{d^*}, K_2^{d^*}, \sigma) \tag{B.1}$$

$$= \mathop{\mathbb{E}}_{F_0} E(d|K_1^{d^*}, K_2^{d^*}, \sigma, F_0) \tag{B.2}$$

$$= \mathop{\mathbb{E}}_{F_0, F_1, F_2} E(d|K_1^{d^*}, K_2^{d^*}, F_1, F_2, F_0) \tag{B.3}$$

$$= \mathop{\mathbb{E}}_{F_0, F_1, F_2} \left[\sum_{i=1,2} \|\hat{F_0} \cdot K_i - F_i\|^2 + \|C \cdot \hat{F_0}\|^2 \right], \tag{B.4}$$

where $\mathbb{E}(x)$ is the expectation of x, and F_i is the i^{th} captured image. Substituting $\hat{F_0}$ with Equation (4), we get:

$$\begin{aligned}
&E(d|K_1^{d^*}, K_2^{d^*}, \sigma) \\
&= \mathop{\mathbb{E}}_{F_0,F_1,F_2} \left[\sum_{i=1,2} \| \frac{F_1 \cdot \bar{K}_1^d + F_2 \cdot \bar{K}_2^d}{|K_1^d|^2 + |K_2^d|^2 + |C|^2} \cdot K_i - F_i \|^2 \right. \\
&\left. + \| C \cdot \frac{F_1 \cdot \bar{K}_1^d + F_2 \cdot \bar{K}_2^d}{|K_1^d|^2 + |K_2^d|^2 + |C|^2} \|^2 \right]. \qquad \text{(B.5)}
\end{aligned}$$

Then, by substituting F_i with Equation (2), we have:

$$\begin{aligned}
&E(d|K_1^{d^*}, K_2^{d^*}, \sigma) \\
&= \mathop{\mathbb{E}}_{F_0,\zeta_1,\zeta_2} \left[\sum_{i=1,2} \| (F_0 \cdot K_i^{d^*} + \zeta_i) - \right. \\
&\frac{(F_0 \cdot K_1^{d^*} + \zeta_1) \cdot \bar{K}_1^d + (F_0 \cdot K_2^{d^*} + \zeta_2) \cdot \bar{K}_2^d}{|K_1^d|^2 + |K_2^d|^2 + |C|^2} \cdot K_i \|^2 + \\
&\left. \| C \cdot \frac{(F_0 \cdot K_1^{d^*} + \zeta_1) \cdot \bar{K}_1^d + (F_0 \cdot K_2^{d^*} + \zeta_2) \cdot \bar{K}_2^d}{|K_1^d|^2 + |K_2^d|^2 + |C|^2} \|^2 \right]. \qquad \text{(B.6)}
\end{aligned}$$

Since ζ_1 and ζ_2 are independent Gaussian white noise $N(0, \sigma)$, we have $\mathbb{E}\,\zeta_i^2 = \sigma^2$, $\mathbb{E}\,\zeta_i = 0$, and $\mathbb{E}\,\zeta_1\zeta_2 = 0$. Let $B = K_1^2 + K_2^2 + C$. Then, Equation B.6 can be rearranged to be:

$$\begin{aligned}
&E(d|K_1^{d^*}, K_2^{d^*}, \sigma) \\
&= \mathop{\mathbb{E}}_{F_0,\zeta_1,\zeta_2} \sum_{i=1,2} \left[\| \frac{F_0[(K_1^{d^*}\bar{K}_1 + K_2^{d^*}\bar{K_2^d}) \cdot K_i^d - K_i^{d^*} B]}{B} \|^2 \right. \\
&\left. + \| \frac{(\zeta_1 \bar{K_1^d} + \zeta_2 \bar{K_2^d}) K_i^d}{B} - \zeta_i \|^2 \right] \\
&+ \| C \cdot \frac{F_0 \cdot (K_1^{d^*} \bar{K_1^d} + K_2^{d^*} \bar{K_2^d})}{B} + \frac{\zeta_1 \cdot \bar{K_1^d} + \zeta_2 \cdot \bar{K_2^d}}{B} \|^2 \\
&= \mathop{\mathbb{E}}_{F_0} \sum_{i=1,2} \| \frac{F_0[(K_1^{d^*} \bar{K_1^d} + K_2^{d^*} \bar{K_2^d}) \cdot K_i^d - K_i^{d^*} B]}{B} \|^2
\end{aligned}$$

$$+\sigma^2\cdot(\|\frac{K_i^{d^2}+C}{B}\|^2+\|\frac{K_1^dK_2^d}{B}\|^2+\|C\cdot\frac{K_i^d}{B}\|^2)$$
$$+\|C\cdot\frac{F_0\cdot(K_1^{d^*}\bar{K_1^d}+K_2^*\bar{K_2^d})}{B}\|^2. \tag{B.7}$$

According to the $1/f$ law, we define the expectation of the power spectrum of F_0 as A, where $A(\xi)=\int_{F_0}|F_0(\xi)|^2\,\mu(F_0)$. In addition, it is known that $C=\sigma^2/A$. Then, Equation B.7 can be further re-arranged and simplified as:

$$\begin{aligned}&E(d|K_1^{d^*},K_2^{d^*},\sigma)\\ =&\sum_\xi\frac{A\cdot|K_1^d\cdot K_2^{d^*}-K_2^d\cdot K_1^{d^*}|^2}{\sum_i|K_i^d|^2+C}\\ +&\sum_\xi\frac{\sigma^2\cdot(\sum_i|K_i^{d^*}|^2+C)}{\sum_i|K_i^d|^2+C}+n\cdot\sigma^2.\end{aligned} \tag{B.8}$$

Appendix C

A Proof of Equation 4.10

When the ratio $c = d/d^*$ approaches to 1, we have

$$\begin{aligned} & M(K_1, K_2, d, d^*) \\ =& \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{(|c-1|d^*)^2 |K_1^{'d^*} K_2^{d^*} - K_2^{'d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2} \right]^{1/2} \\ =& |c-1|d^* \cdot \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1^{'d^*} K_2^{d^*} - K_2^{'d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2} \right]^{1/2}, \end{aligned}$$

where $K_i^{'d^*}$ is the derivative of $K_i^{d^*}$ with respect to the blur size.

Proof: A kernel K can be regarded as a function of both the frequency ξ and the scale d^*. Assume the derivative of K with respect to d^* exists and is denoted by $K^{'d^*}$, we have $K^d = K^{d^*} + \delta d \cdot K^{'d^*}$

when $\delta d = d - d^* = (c-1)d^*$ approaches to zero. Then, we get

$$
\begin{aligned}
& M(K_1, K_2, d, d^*) \qquad \text{(C.1)} \\
=& \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1^d K_2^{d^*} - K_2^d K_1^{d^*}|^2}{|K_1^d|^2 + |K_2^d|^2 + C^2} \right]^{1/2} \\
=& \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|(K_1^{'d^*} + \delta d \cdot K_1^{'d^*}) K_2^{d^*} - (K_2^{d^*} + \delta d \cdot K_2^{'d^*}) K_1^{d^*}|^2}{|K_1^{d^*} + \delta d \cdot K_1^{'d^*}|^2 + |(K_2^{d^*} + \delta d K_2^{d'*})|^2 + C^2} \right]^{1/2} \\
=& |\delta| d^* \cdot \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1^{'d^*} K_2^{d^*} - K_2^{'d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2} \right]^{1/2} . \\
=& |c-1| d^* \cdot \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1^{'d^*} K_2^{d^*} - K_2^{'d^*} K_1^{d^*}|^2}{|K_1^{d^*}|^2 + |K_2^{d^*}|^2 + C^2} \right]^{1/2} .
\end{aligned}
$$

Appendix D

A Proof of Equation 4.11

Consider two scales d_1^* and d_2^* with a ratio $s = d_2^*/d_1^*$, when $c = d/d^*$ approaches to 1, we have

$$M(K_1, K_2, c \cdot d_2^*, d_2^*) \approx M(K_1, K_2, c \cdot d_1^*, d_1^*) \cdot s^{\alpha/2}, \tag{D.1}$$

where α is a constant number that is related to the power order in the $1/f$ law [164].

Proof: According to Equation 4.10, we have

$$\begin{aligned} & M(K_1, K_2, c \cdot d_2^*, d_2^*) \\ =& |c-1| d_2^* \cdot \left[\frac{1}{n} \sum_{\xi} A \cdot \frac{|K_1'^{d_2^*} K_2^{d_2^*} - K_2'^{d_2^*} K_1^{d_2^*}|^2}{|K_1^{d_2^*}|^2 + |K_2^{d_2^*}|^2 + C^2} \right]^{1/2}. \end{aligned}$$

Since K_2^* is a scaled K_1^* of factor s, $K_2^*(\xi) = K_1^*(s\xi)$. Therefore,

$$\begin{aligned} & M(K_1, K_2, c \cdot d_2^*, d_2^*) \\ =& s \cdot |c-1| d_1^* \cdot \left[\frac{1}{n} \sum_{\xi=1}^{n} A(\xi) \cdot \frac{|K_1'^{d_1^*}(s\xi) K_2^{d_1^*}(s\xi) - K_2'^{d_1^*}(s\xi) K_1^{d_1^*}(s\xi)|^2}{|K_1^{d_1^*}(s\xi)|^2 + |K_2^{d_1^*}(s\xi)|^2 + C^2} \right]^{1/2}. \\ =& s \cdot |c-1| d_1^* \cdot \left[\frac{1}{n} \sum_{\eta=s}^{sn} A(\eta/s) \cdot \frac{|K_1'^{d_1^*}(\eta) K_2^{d_1^*}(\eta) - K_2'^{d_1^*}(\eta) K_1^{d_1^*}(\eta)|^2}{|K_1^{d_1^*}(\eta)|^2 + |K_2^{d_1^*}(\eta)|^2 + C^2} \right]^{1/2} \\ =& s \cdot |c-1| d_1^* \cdot \left[\frac{1}{n \cdot s^2} \sum_{\eta=1}^{n} A(\eta/s) \cdot \frac{|K_1'^{d_1^*}(\eta) K_2^{d_1^*}(\eta) - K_2'^{d_1^*}(\eta) K_1^{d_1^*}(\eta)|^2}{|K_1^{d_1^*}(\eta)|^2 + |K_2^{d_1^*}(\eta)|^2 + C^2} \right]^{1/2}, \\ =& |c-1| d_1^* \cdot \left[\frac{1}{n} \sum_{\eta=1}^{n} A(\eta/s) \cdot \frac{|K_1'^{d_1^*}(\eta) K_2^{d_1^*}(\eta) - K_2'^{d_1^*}(\eta) K_1^{d_1^*}(\eta)|^2}{|K_1^{d_1^*}(\eta)|^2 + |K_2^{d_1^*}(\eta)|^2 + C^2} \right]^{1/2}, \end{aligned} \tag{D.2}$$

where $\eta = s\xi$.

According to the $1/f$ law [164], the prior power spectra of natural image $A(\xi)$ statistically takes a form of $D \exp 1/\xi^2$, where the power order may vary slightly around 2 with scenes and D is a normalization factor. This spectra function can be roughly approximated as $A(\xi) = D\frac{1}{\xi^\alpha}$ with a proper α, especially when this prior function is applied to finite-resolution images. Then, $A(\eta/s) \approx A(\eta) \cdot s^\alpha$. Therefore, we have

$$
\begin{aligned}
& M(K_1, K_2, c \cdot d_2^*, d_2^*) \\
\approx & |c-1| d_1^* \cdot \left[\frac{1}{n} \sum_{\eta=1}^{n} A(\eta) \cdot s^\alpha \cdot \frac{|K_1'^{d_1^*}(\eta) K_2^{d_1^*}(\eta) - K_2'^{d_1^*}(\eta) K_1^{d_1^*}(\eta)|^2}{|K_1^{d_1^*}(\eta)|^2 + |K_2^{d_1^*}(\eta)|^2 + C^2} \right]^{1/2}, \\
= & M(K_1, K_2, c \cdot d_1^*, d_1^*) \cdot s^{\alpha/2}.
\end{aligned}
$$

Appendix E

A Proof of Proposition 5.2.1

Theorem E.0.1 *Proposition5.2.1 When an optical diffuser is placed parallel to the sensor plane (see Figure E.1) and the diffusion angle θ is small ($\sin\theta \approx \theta$), we get*

$$\frac{2\tan\theta}{\cos^2\alpha}\cdot\frac{1}{\overline{AB}} = \frac{1}{U} + \frac{1}{Z}, \tag{E.1}$$

where α is the field angle and $\overline{AB}$ is the diffusion size. The perspective projection of P on the diffuser plane C can be approximated with high precision as the center of AB when α is not too large.

Proof: In the following proof, we first use a first order Taylor expansion (the paraxial approximation) to show the DFDiff imaging equation is similar to the Gaussian lens law, and then use a higher order expansion to formulate the image formation more accurately, which proves the proposition.

In Figure E.1, consider the boundary points A and B of the diffusion pattern, we have

$$\begin{cases} \theta_1 - \theta = \theta_2 \\ \theta_3 + \theta = \theta_4 \\ \\ U\cdot\tan\theta_1 + Z\cdot\tan\theta_2 = (U+Z)\cdot\tan\alpha \\ U\cdot\tan\theta_3 + Z\cdot\tan\theta_4 = (U+Z)\cdot\tan\alpha \end{cases} \tag{E.2}$$

I. If $\theta_1, \theta_2, \theta_3$, and θ_4 are very small, the paraxial approximation ($\tan x \approx x$) can be made. With this approximation, we have $\tan(x+y) = \tan x + \tan y$. Thus, $\tan\theta_2 = \tan\theta_1 - \tan\theta$ and $\tan\theta_4 =$

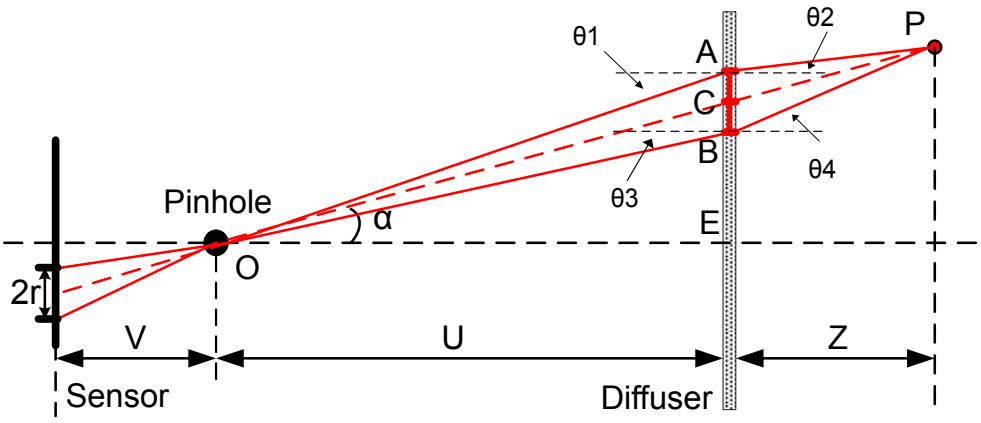


Figure E.1: Geometry of diffusion in a pinhole camera. An optical diffuser with a pillbox diffusion function of degree θ is placed in front of a scene point P and perpendicular to the optical axis. From the viewpoint of pinhole, a diffused pattern AB appears on the diffuser plane.

$\tan\theta_3 + \tan\theta$. The equation group E.2 becomes linear with respect to $\tan\theta_1, \tan\theta_2, \tan\theta_3$ and $\tan\theta_4$. Then, from this equation group, we get:

$$|\tan\theta_1 - \tan\theta_3| = 2\tan\theta \cdot Z/(U+Z) \tag{E.3}$$

and

$$(\tan\theta_1 + \tan\theta_3)/2 = \tan\alpha. \tag{E.4}$$

Therefore, we have

$$\begin{aligned} \overline{AB} &= |\overline{AE} - \overline{BE}| \\ &= U \cdot |\tan\theta_1 - \tan\theta_3| \\ &= 2\tan\theta \cdot UZ/(U+Z), \end{aligned} \tag{E.5}$$

which can also be written as:

$$2\tan\theta \cdot \frac{1}{\overline{AB}} = \frac{1}{U} + \frac{1}{Z}. \tag{E.6}$$

In addition, the center of AB is

$$(\overline{AE} + \overline{BE})/2 = U \cdot (\tan\theta_1 + \tan\theta_3)/2.$$

According to Equation E.4, it is consistent to the perspective projection of P on the diffuser plane, $C = U \cdot \tan\alpha$.

II. Then, instead of using the paraxial approximation (Taylor expansion of ***degree one***), we made a more precise approximation using Taylor expansion of ***degree three*** in order to allow larger field angle α and diffusion angle θ. This gives:

$$\tan(x+y) \approx \tan x + \tan y + \tan x \tan^2 y + \tan^2 x \tan y. \tag{E.7}$$

With this approximation, we can get:

$$|\tan\theta_1 - \tan\theta_3| = \frac{2\tan\theta}{\cos^2\alpha} \cdot \frac{Z}{U+Z}, \tag{E.8}$$

which yields Equation E.1 in Proposition 3.1:

$$\frac{2\tan\theta}{\cos^2\alpha} \cdot \frac{1}{\overline{\overline{AB}}} = \frac{1}{U} + \frac{1}{Z}. \tag{E.9}$$

Also with the approximation, we have:

$$|(\tan\theta_1 + \tan\theta_3)/2 - \tan\alpha| < 0.25\tan\alpha\tan^2\theta, \tag{E.10}$$

which proves the perspective projection of P on the diffuser plane can be approximated with high precision as the center of AB when α and θ are not too large. $\square$

Appendix F

A Proof of Proposition 5.2.1'

Theorem F.0.2 *Proposition5.2.1'**(extended to include tilted diffuser)***

When an optical diffuser is placed parallel to the sensor plane (see Figure E.1) and the diffusion angle θ is small, the size r of the blur pattern on the sensor (radius of the PSF) has:

$$r = V \cdot \frac{\overline{CP}}{\overline{OP}} \cdot \frac{\tan\theta}{\cos^2\alpha}, \tag{F.1}$$

where C is the perspective projection of P on the diffuser.

Remarkably, tilting the diffuser by a small angle will not change the blur size r, as long as the projection C is not changed.

Proof: From Figure E.1, it is easy to see $U = \overline{OC} \cdot \cos\alpha$ and $U + Z = \overline{OP} \cdot \cos\alpha$. Simply

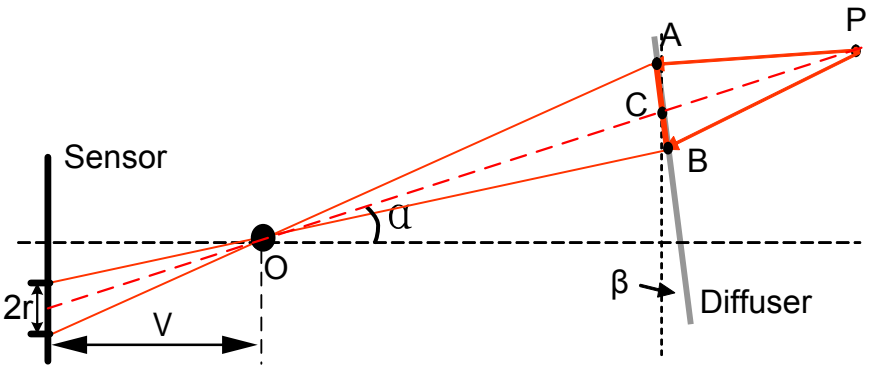


Figure F.1: Geometry of diffusion in a pinhole camera. The diffuser is tilted by a small angle β.

substitute these into Equation E.1, we get:

$$\frac{2\tan\theta}{\cos\alpha}\cdot\frac{1}{\overline{AB}}=\frac{1}{\overline{OC}}+\frac{1}{\overline{CP}}, \tag{F.2}$$

Then, let the diffuser be tilted by a small angle β as shown in Figure F.1. In this case, assume there were a sensor parallel to the diffuser. Then, the conclusions in Proposition 3.1 can still be applied here to compute the size of the diffusion pattern AB on the diffuser. Note that AB indicates the region where the diffused light can reach the pinhole O, and this is obviously independent of the actual placement of the sensor which is behind the pinhole. Since the field position of P with respect to this new sensor plane is $\alpha-\beta$, we have:

$$\frac{2\tan\theta}{\cos(\alpha-\beta)}\cdot\frac{1}{\overline{AB}}=\frac{1}{\overline{OC}}+\frac{1}{\overline{CP}}, \tag{F.3}$$

Therefore,

$$\overline{AB}=\frac{2\tan\theta}{\cos(\alpha-\beta)}\cdot\frac{\overline{OC}\cdot\overline{CP}\cdot}{\overline{OP}} \tag{F.4}$$

Since $\overline{AB}\ll\overline{OC}$ when the diffuser is placed far away from the pinhole, the line AO can be regarded as parallel to BO. With this approximation, the blur size r on the real sensor can be derived as:

$$r=\frac{\overline{AB}}{2}\cdot\frac{V}{\overline{OC}\cdot\cos\alpha}\cdot\frac{\cos(\alpha-\beta)}{\cos\alpha} \tag{F.5}$$

Substitute $\overline{AB}$ using Equation F.4, we get:

$$r=V\cdot\frac{\overline{CP}}{\overline{OP}}\cdot\frac{\tan\theta}{\cos^2\alpha}, \tag{F.6}$$

which is independent of the tilting angle β. This proves Proposition 5.2.1'. □

Part I

Bibliography

Bibliography

[1] JG Ables. Fourier transform photography: a new method for x-ray astronomy. In *Proceedings of the Astronomical Society of Australia*, volume 1, page 172, 1968.

[2] J.E. Adams and JF Hamilton. Design of practical color filter array interpolation algorithmsfor digital cameras. In *Proc. SPIE*, volume 3028, pages 117–125, 1997.

[3] E.H. Adelson and J.R. Bergen. The plenoptic function and the elements of early vision. *Computational Models of Visual Processing*, 1, 1991.

[4] E.H. Adelson and J.Y.A. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):99–106, 2002. ISSN 0162-8828.

[5] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 294–302. ACM, 2004.

[6] M. Aggarwal and N. Ahuja. High dynamic range panoramic imaging. In *IEEE International Conference on Computer Vision*, volume 1, pages 2–9, 2001. ISBN 0769511430.

[7] M. Aggarwal and N. Ahuja. Split aperture imaging for high dynamic range. *International Journal of Computer Vision*, 58(1):7–17, 2004. ISSN 0920-5691.

[8] M. Agrawal and L.S. Davis. Trinocular stereo using shortest paths and the ordering constraint. *International Journal of Computer Vision*, 47(1):43–50, 2002. ISSN 0920-5691.

[9] R.S. Allison, B.J. Gillam, and E. Vecellio. Binocular depth discrimination and estimation beyond interactionspace. *Journal of Vision*, 9(1):10, 2009.

[10] H.C. Andrews and BR Hunt. Digital image restoration. *Prentice-Hall Signal Processing Series, Englewood Cliffs: Prentice-Hall, 1977*, 1977.

[11] S. Baker, T. Sim, and T. Kanade. A characterization of inherent stereo ambiguities. In *IEEE International Conference on Computer Vision*, volume 1, pages 428–435, 2001. ISBN 0769511430.

[12] Y. Bando, B.Y. Chen, and T. Nishita. Extracting depth and matte using a color-filtered aperture. *ACM Transactions on Graphics (TOG)*, 27(5):1–9, 2008. ISSN 0730-0301.

[13] S.T. Barnard and M.A. Fischler. Computational stereo. *ACM Computing Surveys (CSUR)*, 14(4):553–572, 1982.

[14] B.E. Bayer. Color imaging array, July 20 1976. US Patent 3,971,065.

[15] P. Beckmann and A. Spizzichino. The scattering of electromagnetic waves from rough surfaces. *New York*, 1963.

[16] M. Ben-Ezra, A. Zomet, and SK Nayar. Jitter camera: high resolution video from a low resolution detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–135, 2004. ISBN 0769521584.

[17] M. Ben-Ezra, A. Zomet, and S.K. Nayar. Video super-resolution using controlled subpixel detector shifts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 977–987, 2005. ISSN 0162-8828.

[18] Tom Bishop, Sara Zanetti, and Paolo Favaro. Light field superresolution. In *IEEE Conference on Computational Photography*, 2009.

[19] S. Bogner. Introduction to panoramic imaging. In *IEEE SMC Conference*, volume 54, pages 3100–3106, 1995.

[20] M. Born, E. Wolf, and AB Bhatia. *Principles of optics*, volume 10. Pergamon Pr., 1975.

[21] T.E. Boult, X. Gao, R. Micheals, and M. Eckmann. Omni-directional visual surveillance. *Image and Vision Computing*, 22(7):515–534, 2004.

[22] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[23] D.R. Buchele. Unitary catadioptric objective, May 12 1953. US Patent 2,638,033.

[24] JM Burch and C. Forno. A high sensitivity moire grid technique for studying deformation in large objects. *Optical Engineering*, 14:178–185, 1975.

[25] A. Busboom, H.D. Schotten, and H. Elders-Boll. Coded aperture imaging with multiple measurements. *Journal of the Optical Society of America A*, 14(5):1058–1065, 1997. ISSN 1520-8532.

[26] E. Caroli, JB Stephen, G. Cocco, L. Natalucci, and A. Spizzichino. Coded aperture imaging in X-and gamma-ray astronomy. *Space Science Reviews*, 45(3):349–403, 1987. ISSN 0038-6308.

[27] A. Castro and J. Ojeda-Castañeda. Asymmetric phase masks for extended depth of field. *Applied Optics*, 43(17):3474–3479, 2004. ISSN 1539-4522.

[28] W.T. Cathey and E.R. Dowski. New paradigm for imaging systems. *Applied Optics*, 41(29): 6080–6092, 2002. ISSN 1539-4522.

[29] JS Chahl and MV Srinivasan. Reflective surfaces for panoramic imaging. *Applied Optics*, 36 (31):8275–8285, 1997. ISSN 1539-4522.

[30] S. Chang, J. Yoon, H. Kim, J. Kim, B. Lee, and D. Shin. Microlens array diffuser for a light-emitting diode backlight system. *Optics letters*, 31(20):3016–3018, 2006.

[31] D. Chapman and A. Deacon. Panoramic imaging and virtual reality–filling the gaps between the lines. *ISPRS Journal of Photogrammetry and Remote Sensing*, 53(6):311–319, 1998.

[32] J. Charles, R. Reeves, and C. Schur. How to build and use an all-sky camera. *Astronomy Magazine*, 1987.

[33] S. Chaudhuri and AN Rajagopalan. *Depth from defocus: a real aperture imaging approach*. Springer Verlag, 1999. ISBN 0387986359.

[34] T.L. Conroy and J.B. Moore. Resolution invariant surfaces for panoramic vision systems. In *IEEE International Conference on Computer Vision*, page 392, 1999.

[35] O. Cossairt and S. Nayar. Spectral Focal Sweep: Extended depth of field from chromatic aberrations. In *International Conference on Computational Photography*, pages 1–8, 2010.

[36] O. Cossairt, C. Zhou, and S. Nayar. Diffusion coded photography for extended depth of field. In *SIGGRAPH*, pages 1–10. ACM, 2010.

[37] U.R. Dhond and J.K. Aggarwal. Structure from stereo-a review. *IEEE Transactions on Systems, Man and Cybernetics*, 19(6):1489–1510, 1989.

[38] P.L.P. Dillon, AT Brault, JR Horak, E. Garcia, TW Martin, and WA Light. Fabrication and performance of color filter arrays for solid-state imagers. *IEEE Journal of Solid-State Circuits*, 13(1):23–27, 1978. ISSN 0018-9200.

[39] Y. Ding, F. Li, Y. Ji, and J. Yu. Dynamic 3d fluid surface acquisition using a camera array. In *IEEE International Conference on Computer Vision*, 2011.

[40] E.R. Dowski. Passive ranging with an incoherent optical system. 1993.

[41] E.R. Dowski and W.T. Cathey. Extended depth of field through wave-front coding. *Applied Optics*, 34(11):1859–1866, 1995. ISSN 1539-4522.

[42] H. Farid and E.P. Simoncelli. Range estimation by optical differentiation. *JOSAA*, 15(7): 1777–1786, 1998.

[43] P. Favaro and S. Soatto. A geometric approach to shape from defocus. *PAMI*, 27(3):406–417, 2005.

[44] E.E. Fenimore and T.M. Cannon. Coded aperture imaging with uniformly redundant arrays. *Applied Optics*, 17(3):337–347, 1978. ISSN 1539-4522.

[45] C. Forno. Deformation measurement using high resolution moiré photography. *Optics and lasers in engineering*, 8(3-4):189–212, 1988.

[46] C. Gao and N. Ahuja. A refractive camera for acquiring stereo and super-resolution images. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2316–2323, 2006. ISBN 0769525970.

[47] J.M. Geary. *Introduction to lens design: with practical ZEMAX examples*. Willmann-Bell, 2002. ISBN 0943396751.

[48] N. George and W. Chi. Extended depth of field using a logarithmic asphere. *Journal of Optics A: Pure and Applied Optics*, 5:S157, 2003.

[49] T. Georgeiv, K.C. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala. Spatio-Angular Resolution Tradeoff in Integral Photography. In *In Eurographics Symposium on Rendering*, 2006.

[50] T. Georgiev and C. Intwala. Light field camera design for integral view photography. Technical report, Adobe, 2006.

[51] J. Gluckman and S.K. Nayar. Planar catadioptric stereo: Geometry and calibration. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, 1999. ISBN 0769501494.

[52] M. Goesele, B. Curless, and S.M. Seitz. Multi-view stereo revisited. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2402–2409, 2006.

[53] J.W. Goodman. Introduction to fourier optics, mcgaw-hill physical and quantum electronics series, 1968.

[54] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen. The lumigraph. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 43–54. ACM, 1996. ISBN 0897917464.

[55] A. Goshtasby and W.A. Gruver. Design of a single-lens stereo camera system. *Pattern Recognition*, 26(6):923–937, 1993. ISSN 0031-3203.

[56] S.R. Gottesman and EE Fenimore. New family of binary arrays for coded aperture imaging. *Applied Optics*, 28(20):4344–4352, 1989. ISSN 1539-4522.

[57] P.F. Gray. A method of forming optical diffusers of simple known statistical properties. *Journal of Modern Optics*, 25(8):765–775, 1978.

[58] A. Greengard, Y.Y. Schechner, and R. Piestun. Depth from diffracted rotation. *Optics Letters*, 31(2):181–183, 2006. ISSN 1539-4794.

[59] R.D. Guenther. Modern optics. *Modern Optics*, 1, 1990.

[60] F. Guichard, H.P. Nguyen, R. Tessières, M. Pyanet, I. Tarchouna, and F. Cao. Extended depth-of-field using sharpness transport across color channels. *Technical Paper, DXO Labs*, 2009.

[61] B.K. Gunturk, J. Glotzbach, Y. Altunbasak, R.W. Schafer, and R.M. Mersereau. Demosaicking: color filter array interpolation. *IEEE Signal Processing Magazine*, 22(1):44–54, 2005. ISSN 1053-5888.

[62] M. Gupta, Y. Tian, S.G. Narasimhan, L. Zhang, M. Gupta, Y. Tian, S.G. Narasimhan, and L. Zhang. (de) focusing on global light transport for active scene recovery. pages 2969–2976, 2009.

[63] M.G.L. Gustafsson. Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy. *Journal of Microscopy*, 198(2):82–87, 2000.

[64] M.G.L. Gustafsson. Nonlinear structured-illumination microscopy: wide-field fluorescence imaging with theoretically unlimited resolution. *Proceedings of the National Academy of Sciences of the United States of America*, 102(37):13081, 2005.

[65] J.F. Hamilton Jr and J.E. Adams Jr. Adaptive color plan interpolation in single sensor color electronic camera, May 13 1997. US Patent 5,629,734.

[66] R. Hartley. *Multiple view geometry in computer vision*. Cambridge university press, 2008.

[67] S. Hasinoff and K. Kutulakos. Confocal stereo. In *European Conference on Computer Vision*, pages 620–634. Springer, 2006.

[68] S.W. Hasinoff and K.N. Kutulakos. Light-efficient photography. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(1):1, 2009.

[69] S.W. Hasinoff, K.N. Kutulakos, F. Durand, and W.T. Freeman. Time-constrained photography. In *IEEE International Conference on Computer Vision*, pages 333–340, 2009.

[70] G. Hausler. A method to increase the depth of focus by two step image processing. *Optics Communications*, 6(1):38–42, 1972. ISSN 0030-4018.

[71] R.A. Hicks and R.K. Perline. Equiresolution catadioptric sensors. *Applied Optics*, 44(29): 6108–6114, 2005. ISSN 1539-4522.

[72] S. Hiura and T. Matsuyama. Depth measurement by the multi-focus camera. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 953–959, 1998. ISBN 0818684976.

[73] J. Hong, X. Tan, B. Pinette, R. Weiss, and E.M. Riseman. Image-based homing. *IEEE Control Systems Magazine*, 12(1):38–45, 1992.

[74] B.K.P. Horn. Density reconstruction using arbitrary ray-sampling schemes. *Proceedings of the IEEE*, 66(5):551–562, 1978.

[75] G. Indebetouw and H. Bai. Imaging with Fresnel zone pupil masks: extended depth of field. *Applied Optics*, 23(23):4299–4302, 1984. ISSN 1539-4522.

[76] H.E. Ives. Parallax panoramagrams made with a large diameter lens. *Journal of the Optical Society of America A*, 20(6):332–340, 1930.

[77] R. Kingslake. Lens design fundamentals. 1978.

[78] A. Kirmani, T. Hutchison, J. Davis, and R. Raskar. Looking around the corner using transient imaging. In *IEEE International Conference on Computer Vision*, pages 159–166, 2009.

[79] G. Krishnan and SK Nayar. Cata-Fisheye Camera for Panoramic Imaging. In *IEEE Workshop on Applications of Computer Vision*, pages 1–8, 2008.

[80] S. Kuthirummal and S.K. Nayar. Multiview radial catadioptric imaging for scene capture. *ACM Transactions on Graphics (TOG)*, 25(3):916–923, 2006.

[81] S. Kuthirummal and SK Nayar. Flexible Mirror Imaging. In *IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[82] S. Kuthirummal, H. Nagahara, C. Zhou, and S.K. Nayar. Flexible depth of field photography. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):58–71, 2011.

[83] K.N. Kutulakos and S.W. Hasinoff. Focal stack photography: High-performance photography with a conventional camera. *Proc. 11th IAPR Conference on Machine Vision Applications*, pages 332–337, 2009.

[84] L. Larmore. *Introduction to photographic principles*. Dover, 1965.

[85] D.H. Lee, I.S. Kweon, and R. Cipolla. A biprism-stereo camera system. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, 2002. ISBN 0769501494.

[86] M. Lee, FH Bowman, AB Brill, BKP Horn, RC Lanza, JY Park, and CG Sodini. Optical diffusion data acquisition system. *Future Direction of Lasers in Surgery and Medicine, Salt Lake City, Utah*, 1995.

[87] A. Levin. Analyzing Depth from Coded Aperture Sets. In *European Conference on Computer Vision*, 2010.

[88] 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):70–es, 2007. ISSN 0730-0301.

[89] A. Levin, P. Sand, T.S. Cho, F. Durand, and W.T. Freeman. Motion-invariant photography. In *SIGGRAPH*, pages 1–9. ACM, 2008.

[90] A. Levin, S.W. Hasinoff, P. Green, F. Durand, and W.T. Freeman. 4D frequency analysis of computational cameras for depth of field extension. In *SIGGRAPH*, pages 1–14. ACM, 2009.

[91] M. Levoy and P. Hanrahan. Light field rendering. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 31–42. ACM, 1996. ISBN 0897917464.

[92] M. Levoy, R. Ng, A. Adams, M. Footer, and M. Horowitz. Light field microscopy. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 924–934. ACM, 2006.

[93] C.K. Liang, T.H. Lin, B.Y. Wong, C. Liu, and H.H. Chen. Programmable aperture photography: multiplexed light field acquisition. *ACM Transactions on Graphics (TOG)*, 27(3), 2008. ISSN 0730-0301.

[94] S.S. Lin and R. Bajcsy. True single view point cone mirror omni-directional catadioptric system. In *IEEE International Conference on Computer Vision*, volume 2, pages 102–107, 2001. ISBN 0769511430.

[95] G. Lippmann. La photographie int?egrale. *Comptes-Rendus, Acad?emie des Sciences*, (146): 446?–551, 1908.

[96] R. Lukac and K.N. Plataniotis. Color filter arrays: Design and performance analysis. *IEEE Transactions on Consumer Electronics*, 51(4):1260–1267, 2005. ISSN 0098-3063.

[97] A. Lumsdaine and T. Georgiev. The focused plenoptic camera. In *IEEE Conference on Computational Photography*, volume 5, page 6, 2009.

[98] G.M. Mari-Roca, L. Vaughn, J.S. King, K.W. Jelley, A.G. Chen, and G.T. Valliath. Light diffuser for a liquid crystal display, February 14 1995. US Patent 5,390,085.

[99] M. McGuire, W. Matusik, H. Pfister, J.F. Hughes, and F. Durand. Defocus video matting. *ACM Transactions on Graphics (TOG)*, 24(3):567–576, 2005. ISSN 0730-0301.

[100] J.G. McNally, T. Karpova, J. Cooper, and J.A. Conchello. Three-dimensional imaging by deconvolution microscopy. *Methods*, 19(3):373–385, 1999.

[Microsoft] Microsoft. Microsoft kinect depth sensor for xbox. URL `http://www.xbox.com/en-US/kinect`.

[101] M. Mino and Y. Okano. Improvement in the OTF of a defocused optical system through the use of shaded apertures. *Applied Optics*, (10):2219–2225, 1971.

[102] K. Mori. Apparatus for uniform illumination employing light diffuser, July 17 1984. US Patent 4,460,940.

[103] P. Mouroulis. Depth of field extension with spherical optics. *Optics Express*, 16(17):12995–13004, 2008.

[104] D. Mumford and B. Gidas. Stochastic models for generic images. *Quarterly of Applied Mathematics*, (1):85–111, 2001.

[105] H. Nagahara, K. Yoshida, and M. Yachida. An Omnidirectional Vision Sensor with Single View and Constant Resolution. In *IEEE International Conference on Computer Vision*, pages 1–8, 1997.

[106] H. Nagahara, S. Kuthirummal, C. Zhou, and S. Nayar. Flexible depth of field photography. In *European Conference on Computer Vision*, 2008.

[107] S. K. Nayar. Catadioptric omnidirectional camera. In *IEEE Conference on Computer Vision and Pattern Recognition*, page 482, 1997.

[108] S. K. Nayar. Computational cameras: Redefining the image. *Computer*, 39(8):30–38, 2006. ISSN 0018-9162.

[109] S. K. Nayar. Computational camera: Approaches, benefits and limits. *Columbia University, Computer Science Department*, 2011.

[110] S. K. Nayar and V. Branzoi. Adaptive Dynamic Range Imaging: Optical Control of Pixel Exposures Over Space and Time. In *IEEE International Conference on Computer Vision*, page 1168, 2003. ISBN 0769519504.

[111] S. K. Nayar and S. Narasimhan. Assorted pixels: Multi-sampled imaging with structural models. In *European Conference on Computer Vision*, pages 135–315. Springer, 2006.

[112] S. K. Nayar and V. Peri. Folded catadioptric cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, page 2217, 1999.

[113] S. K. Nayar, X.S. Fang, and T. Boult. Separation of reflection components using color and polarization. *International Journal of Computer Vision*, 21(3):163–186, 1997. ISSN 0920-5691.

[114] S. K. Nayar, V. Branzoi, and T.E. Boult. Programmable imaging using a digital micromirror array. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[115] S.K. Nayar and Y. Nakagawa. Shape from focus: An effective approach for rough surfaces. In *IEEE International Conference on Robotics and Automation*, pages 218–225, 1990.

[116] S.K. Nayar, M. Watanabe, and M. Noguchi. Real-time focus range sensor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(12):1186–1198, 1996.

[117] S. A. Nene and S. K. Nayar. Stereo with mirrors. In *IEEE International Conference on Computer Vision*, pages 1087–1094, 1998. ISBN 8173192219.

[118] R. Ng. Digital light field photography. 2006.

[119] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. Light field photography with a hand-held plenoptic camera. *Stanford Computer Science Technical Report*, 2, 2005.

[120] Y. Nomura, L. Zhang, and S. Nayar. Scene collages and flexible camera arrays. In *Proceedings of Eurographics Symposium on Rendering*, 2007.

[121] J. Ojeda-Castaneda and LR Berriel-Valdos. Zone plate for arbitrarily high focal depth. *Applied Optics*, 29(7):994–997, 1990.

[122] J. Ojeda-Castaneda, P. Andres, and A. Diaz. Annular apodizers for low sensitivity to defocus and to spherical aberration. *Optics Letters*, 11(8):487–489, 1986. ISSN 1539-4794.

[123] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 2002. ISSN 0162-8828.

[124] A.P. Pentland. A new sense for depth of field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (4):523–531, 1987. ISSN 0162-8828.

[125] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[126] M. Piana and M. Bertero. Regularized deconvolution of multiple images of the same object. *JOSAA*, 13(7):1516–1523, 1996.

[127] T.C. Poon and M. Motamedi. Optical/digital incoherent image processing for extended depth of field. *Applied Optics*, 26(21):4612–4615, 1987. ISSN 1539-4522.

[128] S. Prasad, T.C. Torgersen, V.P. Pauca, R.J. Plemmons, and J. van der Gracht. Engineering the pupil phase to improve image quality. In *Proc. SPIE*, volume 5108, pages 1–12, 2003.

[129] AN Rajagopalan and S. Chaudhuri. Optimal selection of camera parameters for recovery of depth from defocused images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 219–224, 1997.

[130] R. Raskar, K.H. Tan, R. Feris, J. Yu, and M. Turk. Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. *ACM Transactions on Graphics (TOG)*, 23(3):679–688, 2004.

[131] R. Raskar, A. Agrawal, C.A. Wilson, and A. Veeraraghavan. Glare aware photography: 4D ray sampling for reducing glare effects of camera lenses. *ACM Transactions on Graphics (TOG)*, 27(3):1–10, 2008. ISSN 0730-0301.

[132] A. Rav-Acha and S. Peleg. Restoration of multiple images with motion blur in different directions. In *IEEE Workshop on Applications of Computer Vision*, pages 22–28, 2000.

[133] Stan Reeves. Image deblurring - wiener filter. Blogs, 11 2007.

[134] H. Ren and S.T. Wu. Variable-focus liquid lens. *Optics Express*, 15(10):5931–5936, 2007.

[135] H. Ren, D. Fox, P.A. Anderson, B. Wu, and S.T. Wu. Tunable-focus liquid lens controlled using a servo motor. *Optics Express*, 14(18):8031–8036, 2006.

[136] D. Robinson and D.G. Stork. Joint design of lens systems and digital image processing. In *International Optical Design Conference*. Optical Society of America, 2006.

[137] M.D. Robinson and D.G. Stork. Joint digital-optical design of superresolution multiframe imaging systems. *Applied Optics*, 47(10):B11–B20, 2008.

[138] M. Rouf, R. Mantiuk, W. Heidrich, M. Trentacoste, and C. Lau. Glare encoding of high dynamic range images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 289–296, 2011.

[139] D.L. Rudermant. The statistics of natural images. *Network: Computation in Neural Systems*, pages 517–548, 1994.

[140] J. Salvi, J. Pages, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, 2004.

[141] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado. A state of the art in structured light patterns for surface profilometry. *Pattern Recognition*, 43(8):2666–2680, 2010.

[142] A.B. Samokhin, A.N. Simonov, and M.C. Rombach. Optical system invariant to second-order aberrations. *Journal of the Optical Society of America A*, 26(4):977–984, 2009. ISSN 1520-8532.

[143] Y.Y. Schechner and N. Kiryati. The optimal axial interval in estimating depth from defocus. In *IEEE International Conference on Computer Vision*, pages 843–848, 1993.

[144] Y.Y. Schechner and N. Kiryati. Depth from defocus vs. stereo: How different really are they? *International Journal of Computer Vision*, 39(2):141–162, 2000. ISSN 0920-5691.

[145] Y.Y. Schechner and S.K. Nayar. Generalized mosaicing. In *IEEE International Conference on Computer Vision*, volume 1, pages 17–24, 2001. ISBN 0769511430.

[146] Y.Y. Schechner, S.G. Narasimhan, and S.K. Nayar. Instant Dehazing of Images Using Polarization. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 325–332, 2001.

[147] T. Scheimpflug. Improved method and apparatus for the systematic alteration or distortion of plane pictures and images by means of lenses and mirrors for photography and for other purposes. *GB patent*, 1196, 1904.

[148] C.A. Sciammarella. The moire methoda review. *Experimental Mechanics*, 22(11):418–433, 1982.

[149] R.R. Shannon. The art and science of optical design(book). *Cambridge University Press*, 1997.

[150] J.B. Sibarita. Deconvolution microscopy. *Microscopy Techniques*, pages 1288–1291, 2005.

[151] B.M. Smith, L. Zhang, H. Jin, and A. Agarwala. Light field video stabilization. In *IEEE International Conference on Computer Vision*, pages 341–348, 2009.

[152] H.M. Smith. Light scattering in photographic materials for holography. *Applied Optics*, 11 (1):26–32, 1972.

[153] M. Srinivas and LM Patnaik. Genetic algorithms: a survey. *Computer*, (6):17–26, 1994.

[154] A. Srivastava, AB Lee, EP Simoncelli, and S.C. Zhu. On Advances in Statistical Modeling of Natural Images. *Journal of Mathematical Imaging and Vision*, (1):17–33, 2003.

[155] D.G. Stork and M.D. Robinson. Theoretical foundations for joint digital-optical analysis of electro-optical imaging systems. *Applied Optics*, 47(10):64, 2008. ISSN 0003-6935.

[156] M. Subbarao. Parallel depth recovery by changing camera parameters. In *IEEE International Conference on Computer Vision*, volume 1, 1988. ISBN 0769501494.

[157] M. Subbarao and J.K. Tyan. Selecting the optimal focus measure for autofocusing and depth-from-focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):864–870, 1998.

[158] A. Subramanian, L.R. Iyer, A.L. Abbott, and A.E. Bell. Image Segmentation and Range Sensing Using a Moving Aperture Lens. *IEEE Transactions on Multimedia*, 2001.

[159] G. Surya and M. Subbarao. Depth from defocus by changing camera aperture: A spatial domain approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 61–67, 1993.

[160] E.V. Talvala, A. Adams, M. Horowitz, and M. Levoy. Veiling glare in high dynamic range imaging. In *SIGGRAPH*, pages 37–es. ACM, 2007.

[161] K.H. Tan, H. Hua, and N. Ahuja. Multiview panoramic cameras using mirror pyramids. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):941–946, 2004. ISSN 0162-8828.

[162] H. Tao, H.S. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *IEEE International Conference on Computer Vision*, volume 1, pages 532–539, 2001.

[163] S. Umeyama and G. Godin. Separation of diffuse and specular components of surface reflection by use of polarization and statistical analysis of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):639–647, 2004. ISSN 0162-8828.

[164] A. Van der Schaaf and J.H. Van Hateren. Modelling the power spectra of natural images: statistics and information. *Vision Research*, 36(17):2759–2770, 1996.

[165] C. Varamit and G. Indebetouw. Imaging properties of defocused partitioned pupils. *Journal of the Optical Society of America A*, (6):799–802, 1985.

[166] A. Veeraraghavan, R. Raskar, A. Agrawal, A. Mohan, and J. Tumblin. Dappled photography: Mask enhanced cameras for heterodyned lightfields and coded aperture refocusing. *ACM Transactions on Graphics (TOG)*, 26(3):69, 2007. ISSN 0730-0301.

[167] M. Watanabe and S.K. Nayar. Rational filters for passive depth from defocus. *International Journal of Computer Vision*, 27(3):203–225, 1998. ISSN 0920-5691.

[168] Y. Weiss and W. Freeman. What makes a good model of natural images? In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[169] W.T. Welford. Use of annular apertures to increase focal depth. *Journal of the Optical Society of America A*, 50(8):749–752, 1960.

[170] 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. ISSN 0730-0301.

[171] N. Winters, J. Gaspar, G. Lacey, and J. Santos-Victor. Omni-directional vision for robot navigation. In *IEEE Workshop on Omnidirectional Vision*, pages 21–28, 2000.

[172] L.B. Wolff. Using polarization to separate reflection components. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 363–369, 1989. ISBN 081861952X.

[173] R.J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980.

[174] Y. Xiao and K.B. Lim. A prism-based single-lens stereovision system: From trinocular to multi-ocular. *Image and Vision Computing*, 25(11):1725–1736, 2007. ISSN 0262-8856.

[175] Y. Xiong and S.A. Shafer. Depth from focusing and defocusing. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 68–73, 1993.

[176] K. Yamazawa, Y. Yagi, and M. Yachida. Omnidirectional imaging with hyperboloidal projection. In *International Conference on Intelligent Robots and Systems*, volume 2, pages 1029–1034, 1993. ISBN 0780308239.

[177] J.S. Yedidia, W.T. Freeman, and Y. Weiss. Generalized belief propagation. *Advances in neural information processing systems*, pages 689–695, 2001.

[178] J. Yu and L. McMillan. General linear cameras. In *European Conference on Computer Vision*, pages 14–27. Springer, 2004.

[179] F. Zernike. Diffraction and optical image formation. *Proceedings of the Physical Society*, 61: 158, 1948.

[180] L. Zhang and S. Nayar. Projection defocus analysis for scene capture and image display. *ACM Transactions on Graphics (TOG)*, 25(3):907–915, 2006.

[181] K.C. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala. Spatio-angular resolution tradeoff in integral photography. *Rendering Techniques*, pages 263–272, 2006.

[182] C. Zhou and S. Nayar. What are good apertures for defocus deblurring? In *IEEE Conference on Computational Photography*, pages 1–8, 2009.

[183] C. Zhou and S.K. Nayar. Computational cameras: Convergence of optics and processing. *IEEE Transactions on Image Processing*, 20(12):3322–3340, 2011.

[184] C. Zhou, S. Lin, and S. Nayar. Coded aperture pairs for depth from defocus. In *IEEE International Conference on Computer Vision*, 2009.

[185] C. Zhou, O. Cossairt, and S. Nayar. Depth from Diffusion. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1110–1117, 2010.

[186] A. Zomet and S.K. Nayar. Lensless Imaging with a Controllable Aperture. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 339–346, 2006. ISBN 0769525970.

[187] A. Zomet, D. Feldman, S. Peleg, and D. Weinshall. Mosaicing new views: The crossed-slits projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 741–754, 2003.