A Theory of Depth From Differential Defocus

A Theory of Depth from Differential Defocus

A DISSERTATION PRESENTED
BY
EMMA ALEXANDER
TO
THE DEPARTMENT OF COMPUTER SCIENCE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN THE SUBJECT OF
COMPUTER SCIENCE

HARVARD UNIVERSITY
CAMBRIDGE, MASSACHUSETTS
SEPTEMBER 2018

Thesis advisor: Todd Zickler Emma Alexander

A Theory of Depth from Differential Defocus

Abstract

This document describes a class of computationally-efficient visual depth sensors. Inspired by the visual system of the jumping spider, these sensors are thin lens cameras that observe small changes in optical defocus. Differential defocus changes can be generated by small changes in camera parameters such as aperture size, lens or photosensor location, optical power, camera position, or a combination of these parameters.

The *defocus brightness constancy constraint* describes the resulting differential change in image values I_t as a weighted sum of the spatial derivatives of the blurred image I,

$$I_t = [I_x \quad I_y \quad (xI_x + yI_y) \quad (I_{xx} + I_{yy})]\, \vec{v},$$

where the weights $\vec{v}$ hold information of interest and no knowledge of the underlying scene's texture or geometry is required. This equation is a linear constraint on locally computed image values, so that solving it provides a highly efficient method for recovering depth and other scene information such as velocity.

This dissertation introduces the defocus brightness constancy constraint and describes its usefulness with regard to scene geometry and image content, both analytically and in practice. It also proves the constraint's uniqueness as the only texture-independent linear constraint on local image quantities for differential defocus in a coded aperture camera. The "aperture code" required for the constraint to hold exactly is Gaussian blur, and the method is shown in practice to be robust to nonidealities including non-Gaussian blur, regions with low signal-to-noise ratio, and optically complicated scenes involving reflective and transparent objects. This robustness is demonstrated with a pair of depth from differential defocus sensor prototypes. The first is a standard unactuated camera that observes a moving scene and provides patch-wise depth and velocity measurements. The second contains a lens with electronically-adjustable optical power, and produces per-pixel depth and confidence measurements at 100 frames per second on a laptop GPU.

Contents

Listing of figures

To the women like me who came before, and fought
for a chance to work and a life of freedom and dignity.

Acknowledgments

I have been deeply fortunate to complete this work in excellent company.

Todd Zickler was a wonderful advisor. He is not only a creative and inspiring thinker but also a person of remarkable integrity and humility. His ability to ask simple questions, communicate effectively, and put others before himself have provided a model of the scientist that I hope to become.

Qi Guo, a fellow graduate student who took the lead on the experimental work in this dissertation, is a bright, hard-working, and gentle person who I will always be grateful to know.

Shlomo Gortler was willing to explore widely with me to verify the logic in the proofs in this document. Few people are so generous with their curiosity.

Berthold Horn's published works were an inspiration to this effort, as they have been to so many others. Later, his personal involvement was both a great help and a joy.

J Zachary Gaslowitz, Ioannis Gkioulekas, and Sanjeev Koppal gave their thought and attention to early versions of this work and helped everything get off the ground.

Katie Bouman, Zoya Bylinskii, and Guy Satat provided professional and personal support, both just down the road and around the world.

Paul Shamble was a generous provider of well-cared-for spiders and answers to many questions.

Administrative and custodial staff, especially Kathy Masse, provided practical support and sanity that made this work possible.

Steven W. Zucker, Daniel Holtmann-Rice, the faculty of the Yale physics and computer science departments, the Yale Drop Team, Jeff Rees, and other early mentors gave me the chance to get started and guidance along the way.

I have had the pleasure of sharing my time here with Ying Xiong, Daniel Glasner, Ayan Chakrabarti, Ruonan Li, Gaurav Bharaj, Jialiang Wang, Kathryn Heal, Dor Verbin, Mia Polansky, Anubha Srivastava, and Renbin Liu. Zick's Six will always be my go-to team for interior decorating, even if I'm not allowed to pick the movie again ever.

The support of friends and family runs too wide and deep to tell, but particular gratitude goes to my parents, Ruth and Paul Alexander, for teaching me to value education apparently to the exclusion of a real job; to Bear Flintgruber, Anthea Cheung, Kirklann Lau, Darian Springer, Kathryn Heal, Nicolas Emerson, Julie Woods, and Laura Alexander, for their company in the writing of this document; and to Geva Alexander, for starring in the celebration of its defense.

1
Introduction

Differential changes in optical defocus can reveal scene depth from image pairs through algorithms that require very few adds and multiplies. This document describes the cooperative design of optics and computation to develop sensors based on this depth cue. It includes both a theoretical characterization of the depth from differential defocus cue as well as experimental results from sensor prototypes. This work was inspired by a naturally-occurring depth-specialized camera that has been studied for decades but was recently illuminated by surprising new evidence.

This fascinating camera can be found in the eyes of the jumping spider (Araneae: Salticidae), shown in figure 1.0.1. This common animal's name refers to its ability to jump several times the length of its body to land accurately on targets such as prey or small platforms, demonstrating excellent depth perception that it achieves with a brain the size of a poppy seed. This tiny brain drives remarkably complex behaviors [41] and its visual processing is backed up by an impressive optical system. Spiders have several eyes, and in the jumping spider these can be divided into four to six secondary eyes, which provide low resolution peripheral vision, and two main eyes in front, which have small, high-resolution fields of view that can sweep the scene independently [57]. This anatomy is compatible with stereo or depth from vergence, but behavioral evidence tells a more complicated story.

Behind each of the main eye's lenses, where most animals have a single retina, the jumping spider has a stack of four translucent retinas, each a slightly different distance from the lens [9, 58]. This intricate anatomy has been studied for half a century but its purpose is still not well-understood. The two layers closest to the lens are UV- and polarization-sensitive, and their function remains mysterious, but a recent behavioral study by Nagata et al. [78], summarized below, indicates that the two deepest layers are responsible for the jumping spider's excellent depth perception.

The photosensors in these layers are primarily sensitive to green light, but also slightly sensitive to red light. Chromatic aberration in the corneal lens causes objects that are illuminated with red light to blur as if they were less distant objects illuminated by green light. When the spiders hunt under green light, which their green-sensitive photosensors perceive as similar to normal white illumination, they reliably pounce on their prey, but under red light, they systematically jump short — by exactly the amount that would be predicted from the chromatic aberration of the lens [78]. This is powerful evidence that the jumping spider uses its green-sensitive pair of retinas to perform depth from defocus.

This is very surprising, because the jumping spider is so different from many successful depth from defocus systems in computer vision. A major reason is that its computational resources are extremely limited: the total volume of the spider's brain is comparable to a single hypercolumn in human brain area V1, which we use to preprocess a single pixel of visual input. But artificial depth from

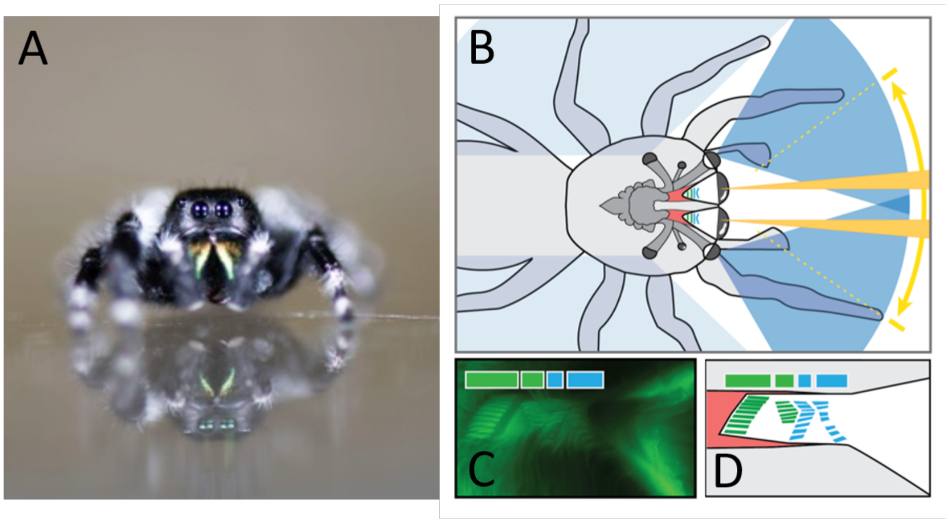


Figure 1.0.1: The jumping spider. Jumping spiders, like the male *Phidippus audax* shown in **A**, have excellent depth perception but a tiny brain. This brain is supported by an impressive optical system, spread across many eyes. These eyes can be separated into six secondary eyes and the two main eyes in front. As shown in **B**, the low-resolution secondary eyes provide a wide field of view peripheral vision system, highlighted in blue, while the high-resolution main eyes have small fields of view that sweep across the scene, shown in yellow. Within the main eye is a stack of four translucent retinas, shown in a confocal microscopy image in **C** and diagrammed in **D**. Due to the difference in their location relative to the lens, these retinas receive images with different amounts of optical defocus. A recent behavioral study has implicated the change in defocus between the back two layers, indicated in green, as the source of the spider's accurate and efficient depth perception. Figure courtesy of Paul Shamble.

defocus algorithms are often very computationally intensive. They also are often assisted by additional hardware, such as coded apertures, that the spider clearly lacks. And, while traditionally the change in defocus is thought of as the source of the depth signal and so designed to be large, the spider seems to experience a small change in defocus across its retinas. These factors suggested that the spider might be performing depth from defocus in a different and highly efficient way, which motivated a search for computational shortcuts.

Indeed, my collaborators and I have found that, under a spider-plausible image formation model, there exists a highly efficient algorithm for extracting scene depth from simple comparisons of image values between and across retinal layers. In this model, the point spread function of the spider's eye is Gaussian, and the width of this Gaussian shrinks or grows as the imaged object moves into or out of focus, so that object depth is encoded by blurriness according to the thin lens equation. This model specifies the image I that would be formed by any textured plane in the world, and by measuring the small change I_t between the retinal layers, and differential changes across a single layer spatially, I_x, I_y, and $\nabla^2_\Sigma I = \alpha_0 I_{xx} + \alpha_1 I_{xy} + \alpha_2 I_{yy}$, depth at the backprojection of each photoreceptor can be measured from a weighted sum of these derivatives:

$$Z = \frac{\nabla^2_\Sigma I}{\beta_0 \nabla^2_\Sigma I + \beta_1 (x I_x + y I_y) + \beta_2 I_t}.$$

The weights $\vec{\alpha}$ and $\vec{\beta}$ are calibrated quantities determined by the dimensions of the eye, and they describe the relative contribution to image change I_t from the magnification change and the defocus change between layers. The magnification change is proportional to the radial derivative — the spatially-weighted directional derivative in the radial direction, $x I_x + y I_y$. For Gaussian blur, the defocus change is proportional to the warped image Laplacian ∇^2_Σ, where weights $\vec{\alpha}$ reflect the covariance of the Gaussian. Solving the equation above in parallel at all pixels gives a depth map of the scene with very few adds and multiplies.

This algorithm may or may not explain the actual mechanism by which the spider measures depth, but regardless of biological accuracy it is not very helpful for building sensors with today's technology. The capacity to manufacture spider-like cameras with stacks of high-quality translucent photosensors at consumer scale does not currently exist and is unlikely to be developed soon.

Fortunately, the specific optical configuration of the spider's eye is far from the only way to measure depth from differential defocus. In addition to sensor location as in the spider's eye, chapter 3 describes how to recover depth from differential changes in: aperture size, as in many classic depth from defocus methods; optical power of the lens, as in an accommodating human eye; lens location, as in a cell phone camera; camera location, as in a moving robot; or from combinations of these changes, as in a cell phone camera held in a moving hand. This describes the capacity of a wide variety of biological and technological sensors illustrated in figure 1.0.2. All of these scenarios generate slightly different equations for depth, but they all follow the same basic paradigm illustrated in figure 1.0.3: given a differentially defocused image pair, take some set of spatial and temporal derivatives, and a physical model provides an equation that extracts depth from the combination of these derivatives.

Specifically, as chapter 3 will show, all of these sensors are governed by a unifying equation for differential defocus change, named the defocus brightness constancy constraint. This equation states that the effect of a differential defocus change on an image can be expressed as a weighted sum of spatial image derivatives:

$$\boxed{I_t = \begin{bmatrix} I_x & I_y & (xI_x + yI_y) & (a_0 I_{xx} + a_1 I_{xy} + a_2 I_{yy}) \end{bmatrix} \vec{v}.}$$

These four coefficients $\vec{v}$ reflect, respectively, the 2D translational motion, magnification change, and defocus change of the sensor.

In the case of the spider (figure 1.0.2.A), there is no translational motion, the magnification change is fixed by the geometry of the eye, and the defocus change reflects depth. In this situation, the first three coefficients are known and the depth recovery equation reduces to the weighted sum of image derivatives previously shown.

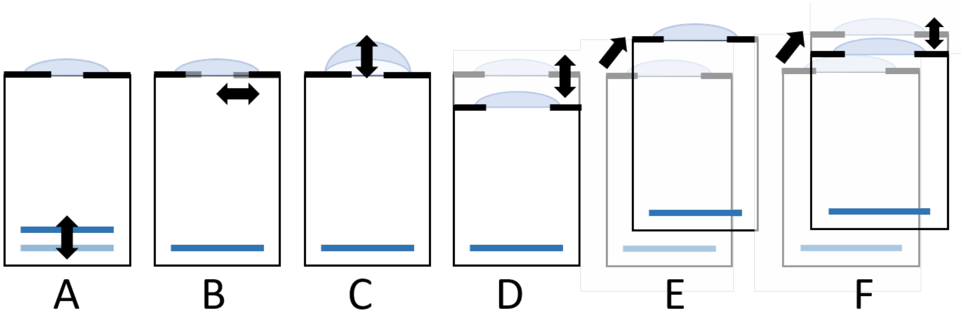


Figure 1.0.2: Mechanisms for differential defocus. Many differential changes to camera parameters cause a defocus change that reveals depth through the defocus brightness constancy constraint. These parameters include: **A** sensor location, as in the spider's eye; **B** aperture width, as in many classic depth from defocus approaches; **C** optical power, as in the focusing of the human eye; **D** lens location, as in a cell phone camera; **E** camera location, as in a sensor mounted on a moving platform or observing a moving scene; and **F** a combination of factors, such as camera and lens location as in a cell phone camera held in a shaking hand. Note that depth is not the only scene property revealed by differential defocus, as velocity can be recovered simultaneously in the relevant scenarios.

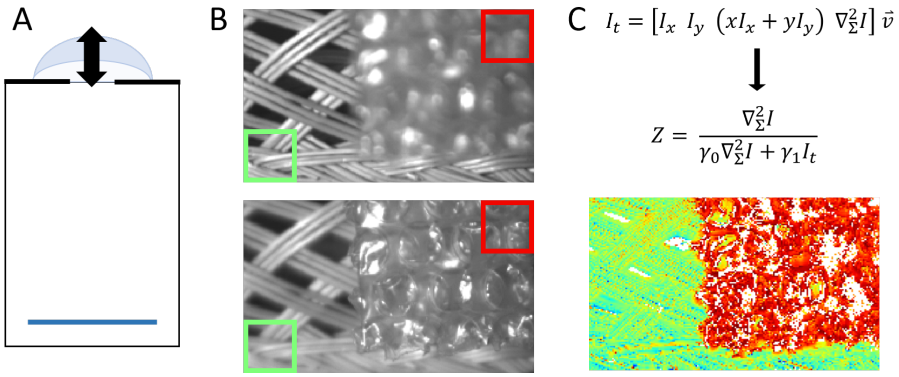

A
B
C
$I_t = [I_x \ I_y \ (xI_x + yI_y) \ \nabla^2_\Sigma I] \ \vec{v}$
$Z = \frac{\nabla^2_\Sigma I}{\gamma_0 \nabla^2_\Sigma I + \gamma_1 I_t}$

Figure 1.0.3: The depth from differential defocus computational pipeline. **A** A camera undergoes a differential change in a parameter that affects defocus, in this case the optical power of the lens (figure 1.0.2.C). **B** A pair of differentially-defocused images is collected. Note the change in blurriness across the image pair in the highlighted regions. **C** An appropriate version of the defocus brightness constancy constraint is used to estimate depth and confidence at each pixel, and depth estimates from high-confidence pixels are reported, while low-confidence regions are shown in white. Note that the optically-challenging bubble wrap is recovered well through this passive, monocular depth cue.

A richer use of this constraint can be seen in the case of camera motion (figure 1.0.2.E). For an unactuated camera mounted on a moving platform or observing a moving scene, the depth and velocity can be measured simultaneously at the backprojection of each image patch by solving a 4×4 matrix equation for the full coefficient vector $\vec{v}$. As will be shown in section 3.2, these weights are one-to-one with depth and 3D camera-to-scene velocity. If the motion is known, this reduces to a per-pixel depth equation as before, but solving for all four coefficients allows depth and 3D velocity to be recovered simultaneously.

The form of this constraint, a linear relationship on local image operations, immediately implies a highly efficient depth sensing algorithm. Section 3.3 proves that for any coded-aperture thin-lens camera, the defocus brightness constancy constraint is the only equation that takes this form. It requires Gaussian blur, and for any other blur kernels, no such local linear constraint can hold exactly. The proof of this uniqueness relies on the theory of distributions, which allows a simultaneous differential analysis of continuous functions and discontinuous objects like binary codes in a consistent way.

Putting the theory into practice requires robustness to both physical and computational non-idealities. Examples of the former include non-Gaussian blur kernels and non-planar, non-Lambertian scenes. In practice, the lenses of the prototypes in chapter 4 have highly non-Gaussian blur but reliably produce high quality depth measurements. This is true even in optically complicated scenes including shiny or transparent objects, such as the bubble wrap shown in figure 1.0.3, demonstrating the advantages of a passive and monocular method like depth from defocus. Examples of the computational challenges include sensor noise, texture-free patches, and finite spatial and temporal resolution and bit depth. Chapter 4 describes the computational methods used to improve robustness with per-measurement confidence estimates, and includes results that accurately measure optically challenging scenes at 100 frames per second on a laptop GPU.

While there remains a great deal of room for the development of differential defocus sensors, this work demonstrates that depth from differential defocus:

1. can be accomplished in many ways,
2. can reveal more than just depth (notably including velocity),
3. can be well-described analytically, and
4. can be made physically and computationally robust without sacrificing efficiency.

The computational efficiency of depth from differential defocus suggests that it could be a valid depth sensing modality for a growing class of microscale sensing platforms, including microrobots and sensor nodes, which have power budgets on the order of milliwatts or even fractions of microwatts [29, 52]. The tight power budgets of these platforms currently put visual sensing out of their reach, but small-scale, task-specific, computationally-efficient vision is happening all around us in the natural world. Understanding how this is done may make vision possible for a new class of microscale devices.

Relation to Previous Publications

This document includes work that has appeared in three prior publications:

[4] Emma Alexander, Qi Guo, Steven J Gortler, and Todd Zickler. Focal flow: Measuring distance and velocity with defocus and differential motion. In *European Conference on Computer Vision (ECCV)*, 2016.

[39] Qi Guo, Emma Alexander, and Todd Zickler. Focal track: Depth and accommodation with oscillating lens deformation. In *International Conference on Computer Vision (ICCV)*, 2017.

[5] Emma Alexander, Qi Guo, Sanjeev Koppal, Steven J Gortler, and Todd Zickler. Focal flow: Velocity and depth from differential defocus through motion. *International Journal of Computer Vision*, pages 1–22, 2017.

Initial work in [4] considers an unactuated, single-retina camera that undergoes uncontrolled motion or observes a moving scene. It develops a limited version of this document's central equation, the defocus brightness constancy constraint, and provides a limited proof of its uniqueness as well as a characterization of inherent sensitivity and results from a proof-of-concept prototype that measures both depth and velocity. In [5] this proof is generalized to something that closely resembles the proof in chapter 3, and experimental results are expanded and improved. Chapter 4 contains experimental work from these papers as well as [39], which describes a fixed camera with a lens whose optical power is fluctuated electronically.

2
Related Work

This work rests on and advances several rich lines of work in the study of depth perception. In addition to the use of defocus in the eye of the jumping spider, it draws from a number of studies on depth from focus and defocus in both natural and artificial vision. The main design approach most closely resembles a line of work in computer vision that can be called differential vision, which extracts scene properties from the observation of small changes to an image that is explicitly modeled as a continuous function of the scene.

2.1 Focus and Defocus in Biological Vision

Jumping spiders are far from the only animals that use optical defocus to better understand their world. Their small brains and complex eyes place tight restrictions on how they might collect and process defocus information, which provides a valuable model for designing power-efficient depth sensors, but there are many other biological systems for depth from focus and defocus.

Many animals can change where their eye is focused, by reshaping or moving a lens, in a process known as accommodation. Chapter 4 describes a prototype that recovers depth from defocus using a differentially-accommodating actuated lens. Accommodation has been shown to provide depth information to toads [27], chameleons [40], and barn owls [112], so that putting lenses on the eyes of these creatures will cause them to systematically misjudge the location of food during tongue extension or pecking. Humans also accommodate and use defocus to determine depth, but chromatic aberration contributes more to both depth perception and accommodation than microfluctuations of the eye [25, 125]. This suggests that the underlying algorithm for human depth from defocus differs from the differential monochromatic methods described here, which do not consider chromatic aberration.

The spider's eye is unable to accommodate and uses a multilayer retina to observe a defocus change instead. This structure is unusual but not unique. Scallops' small eyes feature two photodetecting layers [56], and multiretinal tubular eyes are found in several families of deep sea fish [28]. Of the twenty-eight separate and poorly understood retinas of the sunburst diving beetle larva, all but two are found within twelve multi-layer eyes [71]. Several of these retina stacks lie behind bifocal lenses [100], further complicating defocus information.

Even without these exotic changes to eye structure, optimized defocus blur is found throughout the animal kingdom in the routine specialization of pupil shape. Banks et al. show in [8] that pupil shape is highly correlated with ecological niche: animals with vertical pupils are likely to be ambush predators, while horizontal pupils are often found in prey animals. They point out that elongated pupils allow greater light efficiency while minimizing blur on contours parallel to the pupil, and argue that ambush predators need sharp vertical contours to judge depth from stereopsis, while prey animals benefit from a

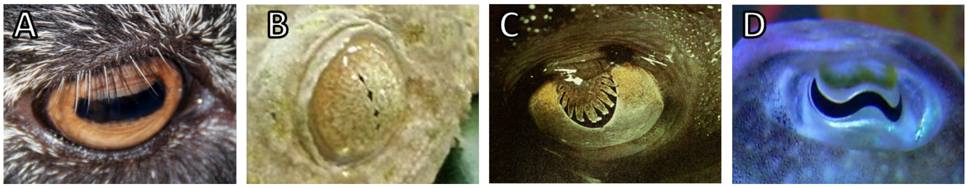


Figure 2.1.1: A variety of natural pupil shapes. Animal pupils display a wide variety of shapes, evolved to suit their visual tasks. **A** shows the eye of a goat. Its horizontal pupil is typical of a prey species, in contrast to the vertical pupils of ambush predators like housecats. **B** shows a gecko pupil, contracted to its four-pinhole state. Several unconfirmed explanations for this pupil shape have been proposed. **C** shows the eye of a skate, a relative of the stingray. This ornate pupil shape has not been explained. **D** shows the W-shaped pupil of a cuttlefish. Like octopi and other cephalopods, they display colorful skin patterns, and this pupil shape is thought to contribute to their color perception.

horizontal panorama that assists in predator detection and navigation of uneven terrain. In many animals, from horses and sheep [8] to snakes, turtles and crocodiles [42], the eye maintains constant pupil orientation even as head position changes, highlighting the importance of appropriate blur kernels to these creatures.

Natural pupil shapes exhibit far more variety than circles and elongated slits. On land, some frogs show heart-shaped or triangular pupils, and keyhole-shaped pupils appear in vine snakes. Under bright light, gecko pupils shrink from round or slit apertures to four pinholes, for unknown reasons perhaps including depth perception [77], reduction of chromatic aberration [54], or camouflage [92]. Under water, exotic off-axis pupils are common among dolphins, rays, and skates. Notably, cephalopods such as octopi and cuttlefish demonstrate a remarkable ability to camouflage themselves by matching their skin to background colors, despite possessing only a single class of photoreceptor in their eyes, a typical indicator of colorblindness. It is thought that their color perception is enabled through an unknown mechanism that relies on their unusual U- or W-shaped pupils [101]. See figure 2.1.1 for an illustration of some of these pupils.

The cooperative development of optical systems and visual algorithms, inherent to the evolution of every biological vision system and highlighted by the task-specific sensing of small-brained animals like the jumping spider, is known in artificial vision as computational sensing or computational photography. Within this field, pupil designs are referred to as aperture codes and a brief summary of their use is included in the following section. Central to the work described in this document is a pupil design problem addressed in chapter 3.

2.2 Focus and Defocus in Computer Vision

Optical focus has been used as a depth cue in artificial vision for decades. When many images are collected under a variety of calibrated camera settings, a search for the most-in-focus image for every patch will yield depth [38]. This approach is called depth from focus, and it is reliable but expensive in terms of time and images captured. It is closely related to the problem of contrast-based autofocus, which selects the most in-focus setting for a single patch of interest. A defining component of both of these problems is the method of determining how in-focus an image patch is.

A review by Pertuz et al. [87] compares many focus measures developed for shape from focus and autofocus in various imaging settings, and clusters them into several major categories. Gradient-based operators [31, 36, 79] observe that full-contrast patches have larger gradient values than blurred patches, Laplacian-based operators [3, 6, 62, 80, 84, 94, 104, 109] similarly measure edge strength by the second order spatial derivatives, wavelet-based measures [121, 122] and discrete cosine transform measures [7, 63, 64, 98] summarize local frequency content, and statistics-based measures rely on image statistics such as Chebyshev moments [116, 123], eigenvalues of image covariance [115–117], variance of gray levels [44, 49, 70, 75, 84], histogram range [33, 95, 106], and histogram entropy [23, 55]. Pertuz et al. conclude that Laplacian focus measures provide the best results for normal imaging conditions but that imaging and texture parameters have a strong effect on a measure's performance for a given focal stack, in a way that is difficult to predict.

Current work on depth from focus includes the ongoing development of focus measures and their application to data from novel devices. Notably, Suwajanakorn et al. in [107] describe a depth from focus pipeline for handheld cameras and mobile phones. While their method requires too much computation to run on current phones, it is able to handle the practical challenges of camera motion, calibration, and the bright image regions caused by blurred light sources, which violate standard assumptions for depth from focus methods.

When restricted to a few images, none of which are guaranteed to be in focus, a depth from defocus algorithm must be used. This approach is more challenging than a depth-from-focus-style search for the sharpest patch because the underlying texture is unknown: a sensor cannot tell if it is capturing a blurry picture of an oil painting or the sharp image of a watercolor. Pentland proposed the first depth from defocus method in [85], noting that the relative attenuation of image frequencies in a defocused image pair could disentangle depth and texture information. These image frequencies can be estimated using spatial filters, such as the narrowband Laplacian filters used by Pentland or by Watanabe and Nayar's carefully-designed broadband filters in [114], or from local Fourier transforms as in [102, 105]. Other algorithms to detect depth from defocus, discussed in further detail below, include: optimizing the deconvolved image for some metric on image statistics [14, 66, 110, 131], solving a stereo problem based on off-axis apertures [61, 96, 111], and computationally blurring images to match each other [30, 108, 120]. This document describes another approach, similar to that of Farid and Simoncelli in [32] or Subbarao and Surya in [103], where spatial and temporal derivative filters reveal depth from image values with high computational efficiency.

Like the specialized animal pupils described in the previous section, depth from defocus sensor performance improves when well-designed attenuation patterns are included in the aperture plane. These are called aperture codes and their design can be optimized for various imaging and vision tasks. Some codes are designed to improve image quality directly at capture time [76, 118] or with a deblurring algorithm [130]. Of particular interest are the following aperture code pairs, designed to both recover depth and improve image quality in cooperation with some postprocessing algorithm.

Zhou and Nayar in [131] present a pair of half-ring aperture codes optimized for statistical recovery of both a depth map and an all-in-focus image. In [65] Levin shows that these half-rings provide near-optimal depth discrimination for aperture code pairs, and that annular rings provide near-optimal performance when more than two non-overlapping codes are allowed. The half-rings, shown in figure 2.2.1.A and B, resemble more light-efficient versions of two-pinhole patterns used for depth from defocus. This set-up is used by Schechner and Kiryati in [96] to describe the fundamental similarity between depth from defocus and small-baseline stereo, in which a small aperture at either edge of a lens can be seen as a stereo camera pair where blur kernel size becomes analogous to stereo baseline. Section 3.2.2 extends this analysis to include depth from differential defocus, showing that inverse magnification serves the analogous role.

Recent applications of the defocus-as-stereo approach can be seen in many phase-based autofocus systems, and in the single-shot systems of Wadhwa et al. [111] and Lee et al. [61]. Wadhwa et al.'s pipeline simulates a wide aperture's limited depth of field using a depth map generated from dual pixels on small-aperture mobile phones. Each dual pixel is split laterally and equipped with a microlens, so that it receives light from half of the aperture, equivalent to a pair of semicircular aperture codes. A stereo algorithm is used to recover depth maps to guide image blurring. Lee et al. present a tricolor pinhole aperture code, shown in figure 2.2.1.C, which generates a depth-dependent offset between the color channels of the captured image. By aligning these color channels with phase correlation [35], scene depth is recovered and used to correct the color errors in the original image.

As shown by Lee et al., a single exposure can provide both depth and a simulated large depth-of-field image when an aperture code is paired with an appropriate inference algorithm. Levin et al. do so in [66] with a binary code, shown in figure 2.2.1.D, designed so that the structure of zeros in the frequency domain reveal blur kernel scale and thereby object depth. The image is retrieved

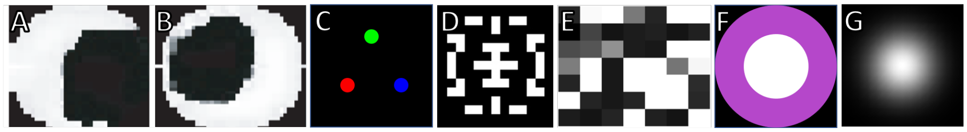


Figure 2.2.1: Aperture codes for depth from defocus and defocus deblurring. These aperture codes are among the many that have been proposed for improving the performance of depth from defocus methods. Depending on the image model and type of code allowed, different patterns are optimal. Code types shown here include **A** and **B** a complementary pair of codes from Zhou et al. [131], **C** a multispectral code from Lee et al. [61], **D** a binary code from Levin et al. [66], **E** a continuous valued but spatially discontinuous code from Veeraraghavan et al. [110], and **F** a spectral code from Chakrabarti and Zickler [14]. Depth from differential defocus is shown in chapter 3 to be uniquely well-suited to a Gaussian code as shown in **G**, which is monochromatic, continous valued, and spatially continuous.

through deconvolution with a sparse derivatives prior. Veeraraghavan et al. in [110] consider a wide class of possible cameras. Their analysis of a standard coded aperture camera optimized for broadband frequencies shows that a continuous-valued but highly pixelated code, shown in figure 2.2.1.E, can be used with a kurtosis-based error function to estimate the blur kernel scale for multilayer scenes, and from this estimated depth map an image is recovered using the digital photomontage techniques of [2]. Chakrabarti and Zickler in [14] consider an RGB image taken through an aperture with a purple ring at the edge, shown in figure 2.2.1.F, so that the green channel is blurred less than the blue and red channels. They show that depth can be recovered from such an image with a prior enforcing agreement of gradient profiles across color channels. Their image recovery, based on efficient deconvolution extending [53], benefits from the increased light efficiency of this aperture code.

The Gaussian filter, shown in figure 2.2.1.G and proven in chapter 3 to be uniquely well-suited to depth from differential defocus, differs significantly from these aperture codes. It is monochromatic, continuous-valued, and spatially continuous. These properties result directly from design choices made in chapter 3, which considers a monochromatic camera (significantly weakening the effect of filters C and F) with a single aperture filter (ruling out filters A and B), then specifies a highly-restricted constraint form and requires that this constraint hold exactly and for any texture. An immediate consequence of the exactness and universality requirement is that it rules out zero crossings in the frequency spectrum of the filter (see equation 3.77). This forbids both the scale signature approach used in designing filter D and the spatial discretization typically used to make filter optimization problems like those resulting in filters A, B, D, and E computationally tractable. Gaussian uniqueness arises as a consequence of algorithm design choices that prioritize computational efficiency over the statistical robustness offered by other coded aperture approaches.

Because of these differences, and due to the fact that its use is rarely derived directly as it is in chapter 3, Gaussian blur is not generally considered a coded aperture model, though it can be expressed as one. And in practice the prototypes of chapter 4 show that, under the optical non-idealities of a real single lens system, a bare lens is roughly as effective as one equipped with a Gaussian-transmittance filter. However, the coded aperture approach greatly influenced this work. The derivation in chapter 3 of the defocus brightness constancy constraint results from an aperture code design problem. Using the theory of distributions, it proves that the Gaussian is the only blur kernel that enables depth sensing from differential defocus with a local linear constraint, out of a large class of aperture filters including binary and pixelated codes as well as continuous functions.

Another unusual feature of depth from differential defocus is its use of small changes in blur. Section 2.3 details three analyses [32, 50, 102] that also consider differential blur change, but such approaches are rare. More common is the recommendation of Rajagopalan and Chaudhuri [88] and Zhou et al. [131] that aperture size should change by a factor of around 1.7 between captures. Tang et al. in [108] likewise roughly double their blur kernel scale between captures, following the analysis of Schechner and Kiryati in [96]. These results are all based on the frequency response of pillbox kernels, however, and so may not be expected to hold for Gaussian blur.

Of these systems, Tang et al. [108] is particularly interesting because they process image pairs from cell phone cameras using a potentially-efficient reblurring algorithm, highlighting the possible usefulness of depth from defocus on restricted platforms. Their reblurring approach, similar to methods seen in [30, 120], convolves each collected image with a bank of filters. These filters are calibrated so that some pair will cause the reblurred image pair to match, and the identity of this kernel pair reveals depth. Tang et al.'s normalized kernels alleviate frequency-dependent bias, and their computationally intensive pipeline includes explicit modeling of scene deformation and a densification of the depth map with several scene priors. They produce high quality full-resolution results of fine scale structure from cell phone image pairs in under an hour.

It should be noted that the difference in runtime between Tang et al.'s method and the systems in chapter 4, which are orders of magnitude faster, is not due primarily to the depth measurement step but instead to their use of priors, deformation modeling, and densification of the depth map. The prototypes described in this document only report sparse, high-confidence measurements. Scene deformation is measured by a passive prototype described in section 4.1 and ignored by a fast (100 *Hz*) actuated prototype in section 4.2. The underlying depth sensing methods seem to be generally comparable in speed and performance.

Given its potential efficiency, reblurring is another plausible hypothesis for the algorithm used by the jumping spider. Other methods based on the application of local filters such as Pentland's [85] or Watanabe and Nayar's [114] hold similar promise. The only way to determine which computations are performed by these animals is to analyze neurological readings from live brains, which are extremely challenging to collect from jumping spiders. Their pressurized bodies quickly fail when the exoskeleton is pierced, and very little work in this area has succeeded. Recent methodological advances in this direction [73] promise valuable insight into the many hyperefficient algorithms the spider must use to perform its complex behaviors.

2.3 Differential Vision

This work continues a line of research that extracts scene information from differential changes in continuously-modeled images. Notable work in this direction includes differential measurements of optical flow [45, 69, 89], time to contact and bearings [46, 47], egomotion [90], depth from photometric stereo [26], depth from defocus [32, 102], shape from shading [11, 126], shape from specularity [1, 13, 82, 128, 129], and simultaneous recovery of depth, lighting, and material for scenes of varying complexity [15–21].

Differential brightness constancy constraints for optical flow are perhaps the most widely known example of differential vision. This classic problem is still an active area of study; for an example of recent work in this field along with a review of previous differential approaches to the problem, see [89]. These approaches consider a model of a pinhole camera observing a moving but constant-brightness scene. One physical quantity that can be extracted from optical flow is time to contact, which is the amount of time before the scene and camera would meet at the center of projection if all velocities remained constant [48, 60]. This quantity can be calculated from the spatial derivatives of the optical flow field, but a more stable method is to consider a 3-term version of the differential brightness constancy constraint, as demonstrated in [46]. This approach provides time to contact and bearings relative to a front-parallel planar scene, with a fast and stable computation and without any calibrated parameters. The defocus brightness constancy constraint developed in this document can be seen as an extension of optical flow constraints that includes the effect of brightness change due to defocus, thereby enabling time to contact and direction of motion to be resolved into depth and, when applicable, 3D velocity.

While defocus or some other depth cue is needed to resolve 3D velocity, there are advantages to only measuring relative velocity, or egomotion, from pinhole images. These include independence from calibration [46] and the ability to generalize to slanted surfaces [47]. Many egomotion methods take a differential approach. A review by Raudies and Nuemann [90] provides a comprehensive comparison of several differential approaches to the problem of egomotion for a variety of scene and motion types under both Gaussian and outlier noise models. They unite five constraints under two underlying scene models (Longuet-Higgins and Prazdny's model of visual image motion [68] for [12, 24, 43, 59, 83, 86, 91, 127], or Longuet-Higgins's stereo model [67] for [51, 132]), categorize the complexity of their optimization techniques (linear for [43, 51, 67, 91], nonlinear for[12, 24, 83, 127], and explicit evaluation of an error function for [59, 86]), and provide guidelines for algorithm selection based on their simulations of each method. Additional camera models have been considered for differential egomotion, notably the large-field-of-view systems in [37, 97], which model the warping of optical flow fields onto curved image planes in order to apply pinhole egomotion algorithms, so these likewise depend on brightness constancy constraints.

The model of a moving camera observing a constant-brightness world draws much of its power from its simplicity, but more complex scenes have also been considered differentially. By considering more complicated models of the camera, scene, and lighting, richer information can be extracted from differential image changes. Examples of this include depth from photometric stereo with a differentially-moving light source [26], shape from shading [11, 126], and shape from differential motion of a reflective object [1, 13, 82, 128, 129].

A series of papers from Chandraker and colleagues provide a thorough theoretical hierarchy of general scene information available from differential observations for several lighting, material, and camera models. They begin by considering light source motion and show that two differential image pairs, under unknown light sources moving on a circle, provide a photometric surface reconstruction for objects of unknown reflectance [19, 20]. Object motion is analyzed in [21], which shows that depth can be recovered with a perspective camera and three object motions (i.e. four images) when the light source is colocated with the camera, or four object motions under unknown lighting. They also specify how many object motions are required to recover more limited scene information (characteristic or level curves) under other lighting and projection models. Camera motion is considered in [15, 17], showing that with three camera motions a perspective camera can recover object depth for objects with a fabric-like appearance under unknown lighting or objects with metal- or plastic-like appearance under known light. Additionally, level or characteristic curves are available to a differentially moving orthographic camera. This is extended to simultaneous recovery of depth and reflectance for dichromatic materials like metals, plastics, and paints in [16], which presents a hierarchy of material types and lighting for scene recovery through differential motion of either the object or the camera. Similar constraints are considered for lightfield cameras in [113], showing that glossy objects with spatially varying reflectance can be constrained so that a quadratic shape prior gives good reconstructions of shape and reflectance. This significant body of work demonstrates the analytical power of differential vision on a wide variety of scenes, and provides a compelling characterization of scene complexity for low-level vision. It does not, however, consider the effects of optical defocus.

Hwang et al. in [50] analyze differential changes in Gaussian-blurred images under changes in photosensor location and object distance. The framing of this work bears significant similarities to section 3.2, except that their model of blur change is not considered simultaneously with magnification change. As a result, their method requires a depth-dependent preprocessing step to correct for magnification, and creates ambiguities in the subsequent computation of depth from defocus that are not physically inevitable.

Early work by Subbarao considers differential changes in defocus in [102]. This work analyzes the Fourier transform of images under simultaneous changes in several camera parameters. It points out the inherent side-of-focal-plane ambiguity in measuring depth from an aperture change, which can only be addressed by focusing outside the range of interest e.g. at infinity, and does not arise for other differential camera changes. Subbarao notes that Gaussian blur provides a neater constraint on depth but suggests using more complicated depth recovery calculations to account for the non-Gaussian blur kernels found in real cameras. This deviation from Gaussian blur is also evident in the prototypes in chapter 4, but the Gaussian model is sufficient to provide good measurements.

Later work by Subbarao and Surya in [103] extends this analysis into the spatial domain. They derive a discrete version of equation 3.16 for locally-cubic textures, radially symmetric blur kernels, and no magnification change. They implement a pair of prototypes, one with a change in aperture size that produces a side-of-focal-plane ambiguity, and one producing non-ambiguous results from a change in lens position. Though their analysis neglects the magnification change generated by this lens motion, in their experiment the change was small and the constraint was still able to recover depth.

Another implementation of depth from differential aperture scaling appears alongside a differential stereo method in [32] from Farid and Simoncelli. They considered differential changes to the scale and center point of a Gaussian aperture filter, as discussed in appendix B. Their sensor consisted of a photosensor and lens, augmented with a small LCD screen which displayed a pair of filters. Rather than taking a pair of images through slightly different Gaussian aperture codes, they directly imaged the scene through the equivalent of a derivative-of-Gaussian filter. This improves the measurement accuracy of image derivatives through what they term optical differentiation.

This document contains three main contributions to previous work on depth from differential defocus: a derivation of the relevant constraint directly from a simple image model in chapter 3, a proof of the uniqueness of Gaussian blur for depth discrimination also in chapter 3, and a pair of prototypes in chapter 4 that demonstrate the speed and robustness available from the differential defocus cue in practice.

3
Theory

The goal of this theory is to describe the changes that occur in an optically defocused image over a differential change in that defocus, without requiring access to an in-focus image of the same scene. A secondary goal is to describe these changes in a way that enables efficient computation of scene depth. This computational efficiency results from a severe restriction on image processing: requiring a linear constraint on locally-computed image quantities, such as spatial derivatives. For a thin lens camera with an aperture code, there is a unifying equation that requires Gaussian blur and describes a large class of possible depth sensors that meet this efficient processing requirement. The equation, named the *defocus brightness constancy constraint*, describes image changes as a weighted sum of spatial image derivatives, where the weights vary by the mechanism of defocus change and are shown to encode scene geometry and motion.

Section 3.1 specifies an image formation model, and section 3.2 shows how differential changes in this image under Gaussian blur can be used to reveal depth. This section derives the defocus brightness constraint for a camera with Gaussian blur kernels, shows how it can be used to recover depth for a large class of sensors, and analyzes the inherent physical sensitivity of differential defocus change as a depth cue. Section 3.3 proves the uniqueness of the Gaussian-based defocus brightness constancy constraint under the specified image processing restrictions.

3.1 Image Formation Model

Image formation is modeled in two steps: the first describes the all-in-focus image P that would be captured by a pinhole camera, in which an object's size and location on the image are determined by its 3D location in the world and the location of the photosensor within the camera. Next, this sharp image is blurred by a convolution with a depth-dependent blur kernel. Specifically, this blur kernel is a stretched and normalized version of an experimenter-designed aperture filter, where the extent of the scaling is determined by a ray optics model of an ideal thin lens.

3.1.1 A Simple World

Assume a camera, with its optical center at the origin of the world, observes a front-parallel plane with a spatially-varying constant-brightness albedo variation, called texture T. Assume that this textured plane has its origin located at a position (X, Y, Z) from the optical center of the camera and that it is projected onto a photosensor an unsigned axial distance f behind the optical center. Note that this distance f will be referred to as the photosensor location rather than the focal length. Focal length is the standard terminology for this quantity in pinhole camera models, but for a camera with a lens, it can also refer to the optical focal length of that lens (the inverse of the lens' optical power). Conflating these two quantities would be equivalent to focusing the camera at infinity, which would constitute a loss of generality. To emphasize their independence, this document will not use the term focal length and instead will refer to photosensor location f and lens optical power p.

3.1.2 All-in-focus Pinhole Image P

First consider the all-in-focus image P that would be captured in this world by a pinhole at the origin. This image,

$$P(x, y) = \eta \; T\left(-\frac{Z}{f}\; x - X, -\frac{Z}{f}\; y - Y\right). \tag{3.1}$$

is simply a magnified and translated version of the scene texture, scaled overall by an exposure-dependent parameter η which can, without loss of generality, be ignored by assuming it is constant and incorporating it into the underlying albedo T.

3.1.3 Optically-Defocused Image I

Now consider replacing the pinhole with an ideal thin lens, as illustrated in figure 3.1.1. For a thin lens with optical power p and photosensor location f, the scene plane will only be imaged in focus if it is at the focal depth Z_f, with

$$p = \frac{1}{f} + \frac{1}{Z_f}, \tag{3.2}$$

and away from this depth it will be blurred by an amount σ determined (e.g. by a similar triangles argument) by scene depth Z according to

$$\sigma = \left(\frac{1}{Z} - \frac{1}{Z_f}\right) f \tag{3.3}$$

$$= \frac{f}{Z} - p\; f + 1. \tag{3.4}$$

In place of the pillbox averaging generally assumed for a thin lens, allow a transmittance profile κ at the aperture, so that the blur kernels k induced on the

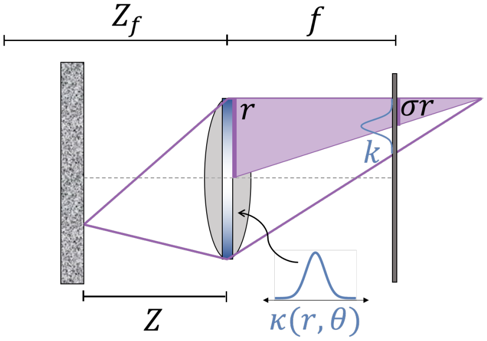


Figure 3.1.1: Image formation model. A front-parallel planar scene at depth Z is imaged by a thin lens. The resulting image I is a blurred version of the all-in-focus image P that would have been collected by a pinhole camera. At the aperture of the lens is a filter with transmittance profile κ, which is an integrable function $\mathbb{R}^2 \to [0,1]$, such as a pillbox, a binary code, or a continuous function. Shown is the Gaussian blur proven in this chapter to be uniquely suitable for depth from differential defocus. The aperture filter κ induces a blur kernel k that is magnified and normalized by a factor σ. This σ is determined by the physical parameters of the system: the location f of the photosensor with respect to the lens, the in-focus depth Z_f (determined by f and the lens' optical power p according to the thin lens law), and the scene depth Z.

image will be replicas of the filter κ, scaled according to the amount of blur σ:

$$k(x,y) = \frac{\kappa\left(\frac{x}{\sigma}, \frac{y}{\sigma}\right)}{\sigma^2}. \tag{3.5}$$

Then the optically-defocused image I is taken to be the underlying pinhole image P, blurred by the depth-dependent blur kernel k:

$$I(x,y) = k(x,y) * P(x,y), \tag{3.6}$$

where $*$ indicates throughout this document convolution in the spatial dimensions x and y. In the following section, differential changes to this image are considered, and Gaussian blur is used to measure depth from the observation of these changes.

3.2 Using Differential Gaussian Defocus Change

A differential change in defocus, when the blur kernels are Gaussian, enables efficient measurement of depth through an equation named the defocus brightness constancy constraint. This section contains a derivation of the constraint (additional derivations are provided in appendix A), shows how to measure depth from Gaussian defocus blur under several kinds of differential camera changes, and analyzes its inherent sensitivity to physical parameters.

3.2.1 The Defocus Brightness Constancy Constraint

Differential changes in defocus blur scale σ become very useful for depth perception when the blur kernels are Gaussian. Let the aperture filter κ be a Gaussian with covariance described by three real numbers a, b, c, with $a, c > 0$ and $4ac > b^2$:

$$\kappa = e^{-ax^2 - bxy - cy^2}. \tag{3.7}$$

Then the induced blur kernels k will be scaled and normalized versions of this filter according to equation 3.5. For a Gaussian filter, this means that a scale change in the blur kernels will be proportional to the sum of their second spatial derivatives, adjusted to the width and orientation of the filter:

$$k_\sigma = \frac{2\sigma}{4ac - b^2} \left(c\, k_{xx} - b\, k_{xy} + a\, k_{yy} \right), \tag{3.8}$$

where subscripts will denote partial derivatives throughout this document. For a radially symmetric Gaussian, with $a = c = 1$ and $b = 0$, this sum of second derivatives will be the filter's Laplacian, $\nabla^2 k = k_{xx} + k_{yy}$. In an abuse of standard terminology, this document will use the term Laplacian to refer to the spatially warped sum $\nabla^2_{\boldsymbol{\Sigma}}$, with

$$\nabla^2_{\boldsymbol{\Sigma}} = c\, \partial_{xx} - b\, \partial_{xy} + a\, \partial_{yy}, \tag{3.9}$$

so that with constant weight Σ^2,

$$\Sigma^2 = \frac{2}{4ac - b^2}, \tag{3.10}$$

the blur kernel change in equation 3.8 can be expressed as a σ-weighted spatial derivative of the kernel itself,

$$k_\sigma = \sigma \Sigma^2 \nabla^2_{\boldsymbol{\Sigma}} k. \tag{3.11}$$

This property of the Gaussian, perhaps best known from scale space analyses [119], reveals scale change from spatial change in a way that can be very useful for depth sensing. Consider a camera that differentially changes the optical power of its lens, so that it observes some known blur scale change $\sigma_t = -f p_t$, while the underlying pinhole image remains the same. In this case, the differential image change over time takes the form

$$\begin{aligned} I_t =& k_t * P \qquad (3.12) \\ =& (k_\sigma \sigma_t) * P \qquad (3.13) \\ =& (\sigma_t \sigma \Sigma^2 \nabla^2_{\boldsymbol{\Sigma}} k) * P. \qquad (3.14) \end{aligned}$$

Of these quantities, only I_t can be measured. However, recall that differentiation commutes with convolution,

$$I_x = k_x * P = k * P_x, \qquad (3.15)$$

and that by the assumption of a front-parallel planar world, the blur change σ_t is spatially invariant and can be pulled out of a spatial convolution. Thus, the unknown blur kernel Laplacian $\nabla^2_{\boldsymbol{\Sigma}} k$ convolved with the unknown pinhole image P gives the measurable image Laplacian $\nabla^2_{\boldsymbol{\Sigma}} I$. It can be seen from this relationship that the ratio of the temporal and spatial change describes depth at the backprojection of each pixel:

$$\begin{aligned} \sigma =& \frac{1}{\sigma_t \Sigma^2} \frac{I_t}{\nabla^2_{\boldsymbol{\Sigma}} I}, \qquad (3.16) \\ Z =& \left(\frac{1}{Z_f} + \frac{\sigma}{f} \right)^{-1}. \qquad (3.17) \end{aligned}$$

Often a defocus change will come with a change in the magnification of the underlying sharp image. Examples of this scenario include a moving object approaching the camera, or the change in photosensor location in the eye of a jumping spider. When defocus change σ_t and underlying image change P_t occur simultaneously, they will both contribute to the observed image change:

$$I_t = k_t * P + k * P_t, \qquad (3.18)$$

and the sources of image change must be disentangled to recover scene information.

As before, the blur change can be related to the image Laplacian,

$$I_t = \sigma_t \, \sigma \, \Sigma^2 \, \nabla_{\boldsymbol{\Sigma}}^2 I + k * P_t, \tag{3.19}$$

where the blur change σ_t can be generated by a change in any of its parameters:

$$\sigma_t = \sigma_p \; p_t + \sigma_Z \; Z_t + \sigma_f \, f_t \tag{3.20}$$

$$= -f \, p_t - \frac{f}{Z^2} \; Z_t + \left(\frac{1}{Z} - p\right) f_t. \tag{3.21}$$

Now consider the new source of image change in equation 3.18, $k * P_t$. The pinhole image will change with any change in the texture location (X, Y, Z) or the photosensor location f:

$$P_t = P_X \; X_t + P_Y \; Y_t + P_Z \; Z_t + P_f \, f_t. \tag{3.22}$$

With texture derivatives indicated by superscripts, this change takes the form

$$P_t = -\, T^{(1,0)} \, X_t - T^{(0,1)} \, Y_t - \left(\frac{x}{f} T^{(1,0)} + \frac{y}{f} T^{(0,1)}\right) \; Z_t + \left(\frac{Zx}{f^2} T^{(1,0)} + \frac{Zy}{f^2} T^{(0,1)}\right) \, f_t. \tag{3.23}$$

Because spatial derivatives of the texture are simply magnified versions of the spatial derivatives of the image,

$$P_x = -\, \frac{Z}{f} T^{(1,0)}, \tag{3.24}$$

$$P_y = -\, \frac{Z}{f} T^{(0,1)}, \tag{3.25}$$

the temporal change P_t of the pinhole image can be described as the following weighted sum of its spatial derivatives:

$$P_t = \frac{f}{Z} \; X_t \; P_x + \frac{f}{Z} \; Y_t \; P_y + \left(\frac{Z_t}{Z} - \frac{f_t}{f}\right) (x P_x + y P_y). \tag{3.26}$$

Note that this is simply the brightness constancy constraint for optical flow, more typically written as

$$P_t = \dot{x} P_x + \dot{y} P_y, \tag{3.27}$$

with

$$\dot{x} = \frac{f}{Z}\, X_t + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) x, \tag{3.28}$$

$$\dot{y} = \frac{f}{Z}\, Y_t + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) y. \tag{3.29}$$

This expanded version of the constraint has the advantage of separating image changes by their cause: the first two terms in equation 3.26 describe image translation, while the last describes a change in magnification with the expression $xP_x + yP_y$. This quantity is the spatially-weighted directional derivative in the radial direction, $\vec{\nabla}_{\vec{r}}$, and it is maximized by image features moving outward from the image center, as when the camera approaches a scene.

For the wide-aperture camera, this change P_t, blurred by a thin lens with Gaussian kernels, is the new source of image change in equation 3.18:

$$k * P_t = k * \left(\frac{f}{Z} X_t\, P_x + \frac{f}{Z} Y_t\, P_y + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) (xP_x + yP_y) \right). \tag{3.30}$$

Again, the spatial invariance assumed of (X, Y, Z) and f and the commutativity of differentiation with convolution allows the first two terms to be expressed with quantities directly computable from the defocused image I,

$$k * P_t = \frac{f}{Z} X_t\, I_x + \frac{f}{Z} Y_t\, I_y + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) (k * (xP_x + yP_y)), \tag{3.31}$$

while the x and y in the third term cannot simply be pulled out of the convolution with the blur kernel k. Instead, note that

$$\begin{aligned} f(x) * (x\,g(x)) = & \int f(X)g(x-X)(x-X)dX \\ =& x\int f(X)g(x-X)dX - \int Xf(X)g(x-X)dX \\ =& x(f(x) * g(x)) - (x\,f(x)) * g(x), \end{aligned} \tag{3.32}$$

so that the blurred radial derivative can be written

$$k * (xP_x + yP_y) = x(k * P_x) - (x\,k) * (P_y) + x(k * P_y) - (y\,k) * (P_y). \tag{3.33}$$

To understand this expression, recall that $I_x = k_x * P = k * P_x$, so that this expression contains the radial derivative of the image,

$$x(k * P_x) + y(k * P_y) = xI_x + yI_y, \tag{3.34}$$

and leaves behind a quantity which, by moving the derivatives across the convolutions, gives an expression,

$$-(x\,k) * (P_x) - (y\,k) * (P_y) = -\,(x\,k)_x * P - (y\,k)_y * P \tag{3.35}$$

$$= -\,(2k + xk_x + yk_y) * P. \tag{3.36}$$

For Gaussian blur specifically, this can be related back to the image Laplacian,

$$-(2k + xk_x + yk_y) * P = \sigma\Sigma^2\nabla^2_\Sigma k * P = \sigma\Sigma^2\nabla^2_\Sigma I, \tag{3.37}$$

so that the blurred radial derivative in equation 3.33 takes the final form

$$k * (xP_x + yP_y) = xI_x + yI_y + \sigma^2\Sigma^2\nabla^2_\Sigma I. \tag{3.38}$$

This completes the manipulations required to describe the change in image value at each pixel over time as a weighted sum of spatial image derivatives, and combining equations3.19, 3.31, and 3.38 immediately produces the defocus brightness constancy constraint:

$$I_t = \begin{bmatrix} I_x & I_y & (xI_x + yI_y) & \nabla^2_{\mathbf{\Sigma}} I \end{bmatrix} \vec{v}, \tag{3.39}$$

with weight vector $\vec{v}$:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} \frac{f}{Z} X_t \\ \frac{f}{Z} Y_t \\ \frac{Z_t}{Z} - \frac{f_t}{f} \\ \left(\sigma_t + \left(\frac{Z_t}{Z} - \frac{f_t}{f}\right) \sigma\right) \sigma \Sigma^2 \end{bmatrix}. \tag{3.40}$$

The values of these weights vary with scene information and can describe a class of sensors to measure depth and, when applicable, velocity.

In the case of the spider, for example, only the photosensor location f changes, so that the defocus brightness constancy constraint has weights

$$\vec{v}_{spider} = \begin{bmatrix} 0 \\ 0 \\ -\frac{f_t}{f} \\ \left(\sigma_f f_t - \frac{f_t}{f}\sigma\right) \sigma \Sigma^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\frac{f_t}{f} \\ -\frac{f_t}{f} \sigma \Sigma^2 \end{bmatrix}, \tag{3.41}$$

which describe the image change,

$$I_t = -\frac{f_t}{f}(xI_x + yI_y) - \frac{f_t}{f} \sigma \Sigma^2 \nabla^2_{\mathbf{\Sigma}} I. \tag{3.42}$$

Note that blur scale σ can be expressed directly in terms of image values:

$$\sigma_{spider} = \frac{I_t + \frac{f_t}{f}(xI_x + yI_y)}{-\frac{f_t}{f} \Sigma^2 \nabla^2_{\mathbf{\Sigma}} I}, \tag{3.43}$$

so that depth at each pixel can be computed from image derivatives and calibrated camera parameters:

$$Z_{spider} = \left(\frac{1}{Z_f} + \frac{\sigma_{spider}}{f} \right)^{-1} = \frac{Z_f \Sigma^2 \nabla_{\mathbf{\Sigma}}^2 I}{\Sigma^2 \nabla_{\mathbf{\Sigma}}^2 I + \frac{Z_f}{f_t} I_t + \frac{Z_f}{f} (x I_x + y I_y)}. \tag{3.44}$$

A more expensive algorithm is needed to compute depth from a defocus change due to camera or object motion. In this case, the per-pixel depth equation is replaced with a per-patch depth and velocity equation. The constraint weight vector takes the form

$$\vec{v}_{motion} = \begin{bmatrix} \frac{f}{Z} X_t \\ \frac{f}{Z} Y_t \\ \frac{Z_t}{Z} \\ \left(\sigma_t + \frac{Z_t}{Z} \sigma \right) \sigma \Sigma^2 \end{bmatrix} = \begin{bmatrix} \frac{f}{Z} X_t \\ \frac{f}{Z} Y_t \\ \frac{1}{Z} Z_t \\ -\frac{f}{Z_f} \frac{Z_t}{Z} \sigma \Sigma^2 \end{bmatrix}. \tag{3.45}$$

Note that unlike the previous scenario, in this case v_4 cannot reveal depth alone, because the quantity σ/Z contains both Z^{-1} and Z^{-2} terms and so is not monotonic in depth. However, the ratio v_4/v_3 is. Specifically,

$$Z_{motion} = \left(\frac{1}{Z_f} - \frac{Z_f}{f^2 \Sigma^2} \frac{v_4}{v_3} \right)^{-1}. \tag{3.46}$$

For unknown motion, depth recovery from v_3 and v_4 requires solving a 4 × 4 matrix multiplication using image derivatives from at least 4 pixels in a patch. However, once depth has been recovered from v_4/v_3, the other three weights reveal 3D velocity:

$$\vec{X}_t = \left(\frac{Z}{f} v_1, \frac{Z}{f} v_2, Z v_3 \right). \tag{3.47}$$

Fundamental computational issues arise when an image patch is degenerate, meaning that the spatial image derivative matrix in equation 3.39 is not full rank. In this case, only partial scene information can be obtained. For example, a patch that contains a single-orientation texture and is subject to the classical aperture problem gives rise to ambiguities in the lateral velocity $(\dot{X}, \dot{Y})$, but depth Z and axial velocity $\dot{Z}$ can still be determined. Separately, in the case of zero axial motion $(\dot{Z} = 0)$, there is no change in defocus and the depth signal is lost. In this case, the third and fourth coefficients will vanish, and the patch can only provide optical flow. In the most extreme case, texture-free image patches will provide no information and neither optical flow nor scene depth can be recovered. However, note that unlike many depth from defocus methods, the combination of magnification and defocus changes makes this method immune to the side-of-focal-plane ambiguity when a defocus change can be measured.

Depth equations for combinations of and restrictions on these models of camera and scene change are straightforward to generate from equation 3.40. The defocus brightness constancy constraint implies a general algorithm for recovering depth and velocity for a large class of actuated or moving cameras: compute spatial and temporal derivatives at each pixel in the patch, solve a least squares estimation for the weight vector $\vec{v}$ that relates these derivatives, and plug these weights into a physical model to recover scene information.

Table 3.2.1 summarizes depth equations for the physical scenarios discussed above. It also includes ambiguous depth recovery available from differential aperture manipulation, such as scaling and rotation analyzed in appendix B.

scenario	change	weights $\vec{v}$	depth recovery
accommodation	p_t	$\left[0, \quad 0, \quad 0, \quad \sigma_p \, p_t \, \sigma\Sigma^2\right]$ $= \left[0, \quad 0, \quad 0, \quad -f \, p_t \, \sigma\Sigma^2\right]$	$Z = \left(\frac{1}{Z_f} - \frac{1}{f^2 p_t \Sigma^2} \frac{I_t}{\nabla_\Sigma^2 I}\right)^{-1}$
spider	f_t	$\left[0, \quad 0, \quad -\frac{f_t}{f}, \quad \sigma_f f_t \, \sigma\Sigma^2 - \frac{f_t}{f}\sigma^2\Sigma^2\right]$ $= \left[0, \quad 0, \quad -\frac{f_t}{f}, \quad -\frac{f_t}{f}\sigma\Sigma^2\right]$	$Z = \left(\frac{1}{Z_f} - \frac{1}{\Sigma^2 f_t} \frac{I_t + \frac{f_t}{f}(xI_x + yI_y)}{\nabla_\Sigma^2 I} + \right)^{-1}$
moving camera	$\vec{X}_t$	$\left[\frac{f}{Z}X_t, \quad \frac{f}{Z}Y_t, \quad \frac{Z_t}{Z}, \quad \sigma_Z \, Z_t \, \sigma\Sigma^2 + \frac{Z_t}{Z}\sigma^2\Sigma^2\right]$ $= \left[\frac{f}{Z}X_t, \quad \frac{f}{Z}Y_t, \quad \frac{Z_t}{Z}, \quad -\frac{Z_t}{Z}\frac{f}{Z_f}\sigma\Sigma^2\right]$	$Z = \left(\frac{1}{Z_f} - \frac{Z_f}{f^2\Sigma^2}\frac{v_4}{v_3}\right)^{-1},$ $\vec{X}_t = \left(\frac{Z}{f}v_1, \frac{Z}{f}v_2, Zv_3\right)$
general hardware	$\vec{X}_t, p_t, f_t$	$\left[\frac{f}{Z} \; X_t, \quad \frac{f}{Z}Y_t, \quad \frac{Z_t}{Z} - \frac{f_t}{f}, \quad \left(\sigma_Z \, Z_t + \sigma_p \, p_t + \sigma_f f_t + \left(\frac{Z_t}{Z} - \frac{f_t}{f}\right)\sigma\right)\sigma\Sigma^2\right]$ $= \left[\frac{f}{Z} \; X_t, \quad \frac{f}{Z}Y_t, \quad \frac{Z_t}{Z} - \frac{f_t}{f}, \quad -\left(f p_t + \frac{f_t}{f} + \frac{f}{Z_f}\frac{Z_t}{Z}\right)\Sigma^2\right]$	$Z = \left(\frac{1}{Z_f} - \frac{1}{f\,\Sigma^2}\frac{v_4}{f\,p_t + (f_t/f) + (fv_3 + f_t)/Z_f}\right)^{-1},$ $\vec{X}_t = \left(\frac{Z}{f}v_1, \; \frac{Z}{f}v_2, \; Z\left(v_3 + \frac{f_t}{f}\right)\right)$
aperture scaling and rotation	a_t, b_t, c_t	See appendix B for details.	$Z = \left(\frac{1}{Z_f} \pm \sqrt{\frac{I_t - (\Delta L)I}{-f^2\nabla_{\Sigma,t}^2 I}}\right)^{-1}$ side of focal plane ambiguity

Table 3.2.1: Applying the defocus brightness constancy constraint. A summary of section 3.2.1. Differential camera changes reveal depth with high computational efficiency using versions of the defocus brightness constancy constraint (equation 3.39).

3.2.2 Inherent Physical Sensitivity of the Constraint

For a camera that observes a blur scale change according to the image model in section 3.1, the defocus brightness constancy constraint is always true, but it will not always be useful. For a trivial example, consider a world plane with no spatial variation in albedo. In this instance any depth from defocus, stereo, or other triangulation-based method will fail. Texture dependence for a given image pair on specific hardware is considered in section 4.2.1. However, it is possible to generalize beyond the reliability of individual depth measurements, to describe the underlying physical sensitivity of the differential defocus cue.

Even in a world with rich texture available at every location and scale, there will come a point where the scene is so blurred that the small changes in image values at a pixel over time and across neighboring pixels will become so small that their comparison is unstable and cannot be used to recover accurate scene information.

This loss of signal away from the in-focus depth is similar to the expected performance of stereo or depth from defocus, for which depth accuracy degrades at large distances. In those cases, accuracy is enhanced by increasing the baseline or aperture size. In differential defocus, inverse magnification plays an analogous role.

Following Schechner and Kiryati in [96], a characterization of responses to a single frequency texture can describe the inherent sensitivity of all three depth cues. Recall that for a stereo system with baseline b and an inference algorithm that estimates disparity Δx, depth is measured as

$$Z = \frac{bf}{\Delta x}, \tag{3.48}$$

with first-order sensitivity to the disparity estimate

$$\left| \frac{dZ}{d(\Delta x)} \right| = \left| \frac{bf}{-(\Delta x)^2} \right| = \frac{Z^2}{bf}. \tag{3.49}$$

Similarly, for a depth from defocus sensor with aperture radius A and an

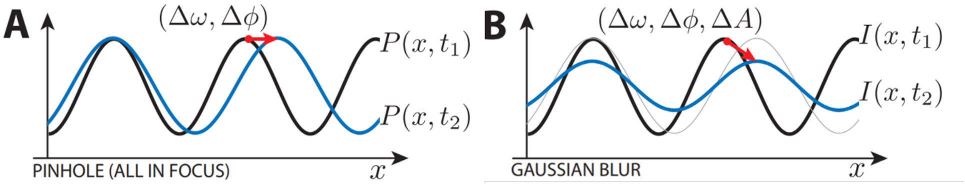


Figure 3.2.1: Observation of a single-frequency texture. A When a 1D pinhole camera observes a world plane with sinusoidal texture, the image is also a sinusoid (black curve). Motion between the camera and scene causes the sinusoidal image to change in frequency and phase (blue curve), and these two pieces of information reveal time to contact and direction of motion. **B** When a finite-aperture camera images a similar moving scene, the motion additionally induces a change in image amplitude, because the scene moves in or out of focus. This third piece of information resolves depth and scene velocity. Figure courtesy of Todd Zickler.

algorithm that estimates blur radius $\tilde{A}$, the sensitivity of depth to error in $\tilde{A}$ is

$$Z = \frac{Z_f f A}{Z_f \tilde{A} + f A}, \tag{3.50}$$

$$\left|\frac{dZ}{d\tilde{A}}\right| = \left|\frac{-Z_f^2 f A}{(Z_f \tilde{A} + f A)^2}\right| = \frac{Z^2}{Af}. \tag{3.51}$$

These equations show a fundamental similarity between stereo and depth from defocus, in which the baseline and aperture size are analogous.

For a toy model of depth from differential defocus, consider images of a sinusoidal texture under Gaussian blur, illustrated in figure 3.2.1. Blurring only changes the amplitude of the image, while motion and magnification change the frequency and phase. Let the texture have frequency ω_0, amplitude B_0, and arbitrary phase and orientation. Then, the image captured at time t will have frequency ω and amplitude B, which are determined by depth:

$$\omega(t) = \frac{Z}{f}\omega_0, \tag{3.52}$$

$$B(t) = \max_{\varphi}(k * P) \tag{3.53}$$

$$= \iint \frac{e^{\frac{-ax^2-bxy-cy^2}{\sigma^2}}}{\sigma^2} B_0 \cos(\omega(t)x)dxdy \tag{3.54}$$

$$= \frac{2\pi}{\sqrt{4ac-b^2}} B_0 e^{-\frac{c}{2}\Sigma^2\sigma^2\omega^2}. \tag{3.55}$$

Over time, these quantities will vary with changes in depth Z and camera parameters, and their relative change will reveal depth.

Regardless of the mechanism of blur change, depth can be measured from image amplitude, frequency, and their derivatives:

$$Z = \left(\frac{1}{Z_f} + \left(\frac{B_t}{c\Sigma^2 f^2 \omega^2 B}\right) \frac{f}{pf_t + fp_t + \frac{\omega_t}{\omega}\frac{f}{Z_f}}\right)^{-1}. \tag{3.56}$$

When image quantities $(\omega, \omega_t, B, B_t)$ are measured within error bounds $(\varepsilon_\omega, \varepsilon_{\omega_t}, \varepsilon_B, \varepsilon_{B_t})$, a simple propagation of uncertainty bounds the depth error ε_Z:

$$\varepsilon_Z \leq \sqrt{\left(\frac{\partial Z}{\partial \omega}\right)^2 \varepsilon_\omega^2 + \left(\frac{\partial Z}{\partial \omega_t}\right)^2 \varepsilon_{\omega_t}^2 + \left(\frac{\partial Z}{\partial B}\right)^2 \varepsilon_B^2 + \left(\frac{\partial Z}{\partial B_1}\right)^2 \varepsilon_{B_t}^2} \tag{3.57}$$

When no blur change is observed, e.g. in the case of a moving camera with an accommodating lens that compensates to keep the object at a constant focus, this error becomes unbounded as the numerical stability of calculations on a tiny B_t break down. However, for several scenarios, a common quantity can be removed from the error bound radicand, suggesting an analogy to stereo and depth from large-scale defocus. For example, in the spider's eye, where $p_t = Z_t = 0$, the depth equation simplifies to

$$Z = \left(\frac{1}{Z_f} + \frac{B_t}{-c\Sigma^2 f \omega \omega_t B} \right)^{-1}, \tag{3.58}$$

so that the error bound becomes, after some algebraic manipulation,

$$\varepsilon_Z \leq \frac{Z|Z - Z_f|}{Z_f} \sqrt{4\frac{\varepsilon_\omega^2}{\omega^2} + \frac{\varepsilon_B^2}{B^2} + \frac{\varepsilon_{B_t}^2}{B_t^2}}. \tag{3.59}$$

Likewise, for a moving but otherwise unactuated camera, as implemented in section 4.1, equation 3.57 takes the form

$$\varepsilon_Z \leq \frac{Z|Z - Z_f|}{Z_f} \sqrt{\frac{\varepsilon_\omega^2}{\omega^2} + \frac{\varepsilon_{\omega_t}^2}{\omega_t^2} + \frac{\varepsilon_B^2}{B^2} + \frac{\varepsilon_{B_t}^2}{B_t^2}}. \tag{3.60}$$

And finally, for an accommodating lens on an otherwise unchanging camera, as implemented in section 4.2,

$$\varepsilon_Z \leq \frac{Z|Z - Z_f|}{Z_f} \sqrt{4\frac{\varepsilon_\omega^2}{\omega^2} + \frac{\varepsilon_B^2}{B^2} + \frac{\varepsilon_{B_t}^2}{B_t^2}}. \tag{3.61}$$

In each of these expressions, the sum of error terms in the radicand describes the relative usefulness of improving accuracy in either brightness or spatial frequency measurements for a given scene. It could guide the design or selection of an optimized photosensor, e.g. [124], because when combined with an appropriate statistical model of the scenes to be imaged, it quantifies the trade-off between bit depth, which places a lower bound on ε_B and ε_{B_t}, and spatial resolution, which likewise bounds ε_ω and ε_{ω_t}.

Depending on the error model, the radicands in these error bounds could introduce complex texture and scene dependencies, but in the simplest case, they are constant. In this case, differential defocus is immediately comparable to stereo and classic depth from defocus. Just as the sensitivity of those measurements goes as depth squared, depth from differential defocus measurements are sensitive to object distance from both the camera and the focal plane through the $Z|Z - Z_f|$ term. The differential defocus analogue to aperture size or baseline in this scenario is inverse magnification, the ratio of in-focus depth Z_f to photosensor location f.

3.3 Uniqueness of the Defocus Brightness Constancy Constraint

The family of constraints analyzed above relies on the use of Gaussian blur. This is more than just a convenient assumption; it turns out to be the only set of blur kernels that enables differential depth measurement from a blur scale change using local post-processing operations. In this section, depth from differential defocus is cast as a coded aperture problem, with the result of a theorem proving that the Gaussian is the only code that enables constraints of a specific, computationally-efficient form.

Recall that the image formation model allows the placement of any physical filter κ at the camera's aperture, which will generate blur kernels k that are scaled and normalized versions of that filter. Without the assumption of Gaussian blur, the differential image change over time can be divided into lateral on-image motion that is independent of the blur change, similar to optical flow with a pinhole camera, and the residual brightness change R due to defocus after this motion has been accounted for:

$$I_t = \frac{f}{Z} X_t I_x + \frac{f}{Z} Y_t I_y + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) (x I_x + y I_y) + R. \tag{3.62}$$

The image formation model provides the exact form of this per-pixel residual brightness change R,

$$R = - \left(\frac{Z_t}{Z} - \frac{f_t}{f} + \frac{\sigma_t}{\sigma} \right) (2k + x k_x + y k_y) * P, \tag{3.63}$$

which involves, from left to right: a combination of scene parameters that are known or being measured (Z, Z_t, f, f_t, σ, and σ_t), a function of the blur kernels k generated by the aperture filter, and the underlying pinhole image P, which can only be accessed indirectly through the defocus blurred image I.

The goal is to co-design an aperture filter κ and a depth-blind post-processing operation M such that $M[I]$ is inexpensive to compute (specifically, it is required to be spatially uniform and apply to a sub-infinite number of pixels at any point) and produces a quantity proportional to this residual, without resorting to knowledge of the texture through expressions involving P. As stated in the following theorem, the Gaussian aperture filter and warped Laplacian operator turn out to be the only such pair of objects.

Theorem. *Let k be induced by some $\kappa : \mathbb{R}^2 \to [0,1]$ with $\kappa(x,y)$, $x\kappa(x,y)$, and $y\kappa(x,y)$ Lebesgue integrable and κ not identically zero. For $v \in \mathbb{R}$ and translation-invariant linear spatial operator M with finite support,*

$$v\, M\left[k * P\right] = R(k, P) \tag{3.64}$$

for all compactly supported P, if and only if there are constants $c_1 \in (0,1]$, $c_2 \in \{\mathbb{R} - 0\}$ and a real symmetric positive definite matrix $\mathbf{\Sigma}$ such that

$$\kappa = c_1\, e^{-\vec{x}^T \mathbf{\Sigma} \vec{x}}, \tag{3.65}$$

$$M = c_2 \nabla^2_{\mathbf{\Sigma}} = \; c_2 \partial^T_{\vec{x}} \mathbf{\Sigma}^{-1} \partial_{\vec{x}}. \tag{3.66}$$

This theorem states that, when the filter κ is a Gaussian with covariance matrix $\mathbf{\Sigma}$, the residual R is proportional to the image Laplacian of inverse covariance, $M[I] \propto \nabla^2_{\mathbf{\Sigma}} I = \mathbf{\Sigma}^{-1}_{11} I_{xx} + 2\mathbf{\Sigma}^{-1}_{12} I_{xy} + \mathbf{\Sigma}^{-1}_{22} I_{yy}$, which is directly observable from image information. Moreover, the Gaussian is the *only* aperture filter — out of a broad class of possibilities including pillboxes, binary codes, and smooth functions, but excluding pinholes — that permits exact observation by a depth-blind, translation-invariant linear operator.

3.3.1 Mathematical Background

The following proofs draw heavily on the theory of distributions, see [93] as a reference, and on some concepts from complex analysis. This section will briefly introduce the relevant terms and properties before continuing with the proof of the theorem.

The theory of distributions generalizes functional analysis beyond the classic notion of a function. Much of its usefulness comes from extending results from the study of differential equations to include discontinuous objects whose derivatives do not exist in the traditional sense, but can nonetheless be abstracted in consistent and powerful ways. While standard functions and measures can be treated as distributions, so can more exotic objects like the Dirac delta "function", which is the distributional derivative of the discontinuous step function, as well as its derivatives in turn.

In this context, a distribution is defined as a linear functional that maps some set of well-behaved functions to the real numbers. Unlike functions, they do not have values at given points in a domain, though this can be a useful way to visualize their effect. Any locally-integrable function P can induce a distribution $\check{P}$, indicated by an accent mark, that maps a good test function f (more detail below) to the reals through integration:

$$\langle \check{P}, f \rangle = \int\limits_{\mathbb{R}^n} P(\vec{x}) f(\vec{x})\, d\vec{x}, \tag{3.67}$$

while the Dirac delta maps each function to its value at the origin:

$$\langle \delta, f \rangle = f(0). \tag{3.68}$$

Distributions in this sense should not be confused with probability or frequency distributions, distributions as defined in differential geometry, or any of the many other scientific uses of the term.

Many operations require more care in their application to distributions than to functions. While distributions can be added together and multiplied with real numbers or with infinitely differentiable functions, the product of two distributions, for example, is not well-defined. One of the most useful operations that can be performed on a distribution is taking its derivative. This operation is defined by moving the derivative onto the test function (with a sign change), and allows all distributions to be treated as infinitely differentiable with many of the properties of classical derivatives. This allows meaningful use of objects like the n^{th} derivative of the Dirac delta:

$$\langle \delta^{(n)}, f \rangle = (-1)^n \langle \delta, f^{(n)} \rangle = (-1)^n f^{(n)}(0). \tag{3.69}$$

In describing the properties of distributions, it is useful to classify them by the sets of test functions that they handle gracefully. There are many choices of f that could lead equation (3.67) to violate the definition of a distribution, such as any complex-valued function. Typically, test functions are drawn from the space $D(\mathbb{R}^n)$, which is the set of infinitely-differentiable, real-valued, compactly-supported functions. A distribution must linearly map any member of this set to a real number. The space of distributions is called $D'(\mathbb{R}^n)$, as the dual space of $D(\mathbb{R}^n)$.

By considering larger sets of test functions, one can define smaller sets of distributions that still linearly map all allowed test functions to the reals. Two such classes are used in this document. The first is the set of tempered distributions. The test function of a tempered distribution does not have to be compactly supported, but can be any rapidly-decreasing smooth function. The space of these test functions is called Schwartz space or $S(\mathbb{R}^n)$ and notably includes Gaussians and their derivatives. By its integrability and boundedness, the most general form of aperture filter allowed in the proof is a tempered distribution, and tempered distributions are closed under differentiation: $\check{\kappa} \in S'(\mathbb{R}^2)$ implies $\check{\kappa}_x, \check{\kappa}_y \in S'(\mathbb{R}^2)$.

A useful subset of the tempered distributions is the set of distributions with compact support. These distributions map any test function to zero if the support of that function excludes a certain compact region, called the support of the distribution. The Dirac delta is a classic example of a compactly-supported distribution, because any test function with $f(0) = 0$ is mapped to zero, so $\operatorname{supp}(\delta) = \{0\}$. The theorem requires the distributions induced by the post-processing operation m and the pinhole image P to have compact support: $\check{m}, \check{P} \in \mathcal{E}'(\mathbb{R}^2)$.

All of this is relevant because it establishes when the processed image $M[I] = m * k * P$ is well-defined. The convolution theorem, which states that convolution can be performed by multiplication of Fourier transforms, holds for:

1. two L^1 functions, producing another L^1 function.

2. a tempered distribution and a compactly-supported distribution, producing a tempered distribution.

3. a rapidly decreasing function with a tempered distribution, producing another tempered distribution.

The first of these describes the traditional use of the theorem, the second is the reason that P must be compactly supported for general (tempered) κ in the theorem, and the third allows the assumption of compactness on P (which is bounded and locally integrable, so $\check{P}$ is tempered) to be dropped in the use of the constraint after $m * \kappa$ is shown to be a rapidly-decreasing function.

The proof also makes use of Schwartz's Paley-Weiner theorem, which states that the Fourier transform of a compactly-supported distribution on the reals is an entire function. This is a very powerful result in complex analysis, see [34] for a reference. Complex analysis extends analysis to functions on the complex numbers, creating alternate versions of familiar ideas from calculus on the reals. Several of these appear in the proof, particularly in claims 3 and 4.

Perhaps the most important of these concepts is the complex derivative. This is defined, just as on the reals, as the limit of the difference quotient, but it will exist in far fewer cases. Take, for example, the function $\Re(z)$, which returns the real part of its complex input z. Using the standard metaphor of $\mathbb{R}^2$ for $\mathbb{C}^1$, one could imagine this function as having perfectly well-defined partial derivatives: 1 along the real axis, 0 along the imaginary axis. However, because the derivative is a single limit, which must match from all directions of approach in order to exist, the function $\Re(z)$ is in fact nowhere complex differentiable.

As a result of this restrictive definition, differentiable functions are much rarer in complex analysis, and they have a number of special properties that real-differentiable functions lack. Functions that are complex differentiable in a neighborhood, called analytic or holomorphic functions, are, for example, infinitely differentiable everywhere the first derivative exists.

For complex functions that are holomorphic except at isolated points, there are three kinds of singularities that can occur: removable singularities, poles, and essential singularities. A removable singularity is like a patchable hole in the function — the function is not defined at the point, but it can be continuously extended to a function that is. A pole is a point at which the function goes to complex infinity (a quantity with infinite magnitude and indeterminate phase) but where the product of the function and some polynomial is holomorphic at that point. Anything more serious, like an oscillating discontinuity or a non-pole infinity, is called an essential singularity.

A holomorphic function with no singularities at any point other than infinity is called an entire function. These are very special and arise in the proof of claim 4. They include polynomials, exponentials, trigonometric functions, and their sums, products, compositions, derivatives, and integrals. According to Liouville's theorem, any entire function whose magnitude is bounded must be constant, so any non-constant entire function must have a singularity at infinity. If this singularity is essential, the function is transcendental (e.g. sine or cosine) and if it is a pole, the function is a polynomial. This restriction, along with Schwartz's Paley-Weiner theorem, is used to prove claim 4.

3.3.2 Proof of the theorem

Several claims will be used to prove the theorem. Briefly, claim 1 establishes, from the Fourier transform of the residual, a texture-independent relationship between the spectra of the filter and the operator in the form of a differential equation, which can be solved with standard methods. Claim 2 restricts the form of depth-blind operators using a separability argument. Claims 3 and 4 further restrict this form to ensure that the operator uses a sub-infinite number of pixels. Claim 5 enforces the non-negativity of the aperture transmittance to complete the proof.

Begin by noting that the operator $M[I]$ can be expressed as a convolution $m * I$ with some compactly-supported m. While the standard terms for M and m are an operator and a filter, respectively, this document will refer to m as an "operation" rather than a "filter" to emphasize that it is a computational object and to distinguish it from the physical, light-blocking filter κ at the camera's aperture.

Claim 1. *The blur kernel k and post-processing operation m are related in the frequency domain by* $\hat{k}(\hat{r},\hat{\theta}) = f(\hat{\theta})e^{-w\int_0^{\hat{r}}\frac{\hat{m}(s,\hat{\theta})}{s}ds}$ *for* $w(\sigma) = \frac{Z-Z_f}{Z_t}v$ *and some angular function* $f(\hat{\theta})$.

Proof of Claim 1. The Fourier transform takes the convolution

$$v\, m * k * P = R \tag{3.70}$$

to a multiplication

$$v\, \hat{m}\, \hat{k}\, \hat{P} = \hat{R}, \tag{3.71}$$

with hats indicating the Fourier transforms of the original distributions, expressed in polar coordinates $(\hat{r},\hat{\theta}) = \left(\sqrt{\omega_x^2+\omega_y^2}, \tan^{-1}(\omega_x,\omega_y)\right)$.

The Fourier transform of the residual takes the form

$$\begin{aligned}\hat{R} =& \mathcal{F}\left[\frac{Z_t}{Z - Z_f}(2k + xk_x + yk_y) * P\right] && (3.72)\\ =& \frac{Z_t}{Z - Z_f}\,\mathcal{F}[(2k + xk_x + yk_y)]\,\hat{P}, && (3.73)\end{aligned}$$

where

$$\begin{aligned}\mathcal{F}[(2k + xk_x + yk_y)] =& 2\hat{k} + i\partial_{\omega_x}(i\omega_x\hat{k}) + i\partial_{\omega_y}(i\omega_y\hat{k}) && (3.74)\\ =& -\omega_x\hat{k}_{\omega_x} - \omega_y\hat{k}_{\omega_y} && (3.75)\\ =& -\hat{r}\hat{k}_{\hat{r}} && (3.76)\end{aligned}$$

so equation (3.71) can be rewritten as

$$v\,\hat{m}\,\hat{k}\,\hat{P} = -\frac{Z_t}{Z - Z_f}\,\hat{r}\,\hat{k}_{\hat{r}}\,\hat{P}. \qquad (3.77)$$

This is required to hold for all underlying scene textures by dropping the $\hat{P}$ term from either side, leaving a simple partial differential equation on $\hat{k}$. Compactness of m guarantees that $\hat{m}$ is smooth, and integrability of k, xk, and yk guarantee that $\hat{k}$, $\hat{k}_{\omega_x}$, and $\hat{k}_{\omega_y}$ are continuous, so this equation can be solved using integrating factors. □

Claim 2. *The Fourier transform of the post-processing operation, $\hat{m}$, takes the form $g(\hat{\theta})\hat{r}^n$ for some $n \in \mathbb{C}$ and angular function $g(\hat{\theta})$.*

Proof of Claim 2. Recall that the post-processing operation is required to be depth-blind, so $\hat{m}$ cannot be a function of the depth-scaling factor σ. However, equation (3.64) must hold for the entire family of possible blur kernels k, which are depth-scaled versions of the physical aperture filter κ according to equation (3.5). In the frequency domain this depth scaling takes the form

$$\hat{k}(\hat{r},\hat{\theta}) = \hat{\kappa}(\sigma\hat{r},\hat{\theta}). \qquad (3.78)$$

This means that it is possible to introduce the functions

$$\alpha(\sigma\hat{r},\hat{\theta}) = \ln\left(\frac{\hat{\kappa}(\sigma\hat{r},\hat{\theta})}{f(\hat{\theta})}\right), \tag{3.79}$$

$$\beta(\sigma) = -\,w(\sigma), \tag{3.80}$$

$$\gamma(\hat{r},\hat{\theta}) = \int\limits_{0}^{\hat{r}} \frac{\hat{m}(s,\hat{\theta})}{s} ds, \tag{3.81}$$

and rewrite a slightly rearranged form of claim 1 as

$$\alpha(\sigma\hat{r},\hat{\theta}) = \beta(\sigma)\gamma(\hat{r},\hat{\theta}). \tag{3.82}$$

Considering what happens when $\hat{r} = 1$, see that

$$\alpha(\sigma,\hat{\theta}) = \beta(\sigma)\gamma(1,\hat{\theta}), \tag{3.83}$$

so α is separable in $\hat{\theta}$. This separability can be seen in the general-$\hat{r}$ case by replacing σ with $\sigma\hat{r}$ for an alternate expression for α:

$$\alpha(\sigma\hat{r},\hat{\theta}) = \beta(\sigma\hat{r})\gamma(1,\hat{\theta}). \tag{3.84}$$

Taking the $\hat{r}$ derivative of equations (3.82) and (3.84) and noting that they must be equal, gives

$$\frac{d}{d\hat{r}}\big(\beta(\sigma)\gamma(\hat{r},\hat{\theta})\big) = \beta(\sigma)\gamma^{(1,0)}(\hat{r},\hat{\theta}) \tag{3.85}$$

$$= \frac{d}{d\hat{r}}\big(\beta(\sigma\hat{r})\gamma(1,\hat{\theta})\big) = \sigma\beta'(\sigma\hat{r})\gamma(1,\hat{\theta}), \tag{3.86}$$

so that again considering the $\hat{r} = 1$ case, it can be seen that

$$\beta(\sigma) = \frac{\gamma(1,\hat{\theta})}{\gamma^{(1,0)}(1,\hat{\theta})}\sigma\beta'(\sigma). \tag{3.87}$$

This is a separable ordinary differential equation that has the solution

$$\beta(\sigma) = c\sigma^{\frac{\gamma^{(1,0)}(1,\hat{\theta})}{\gamma(1,\hat{\theta})}} \tag{3.88}$$

for some constant c. Because $\beta(\sigma)$ cannot change with $\hat{\theta}$, the exponent must also be a constant n. These forms of β and n allow equation (3.86) to be rewritten

$$\begin{aligned} \gamma^{(1,0)}(\hat{r},\hat{\theta}) &= \frac{\sigma\beta'(\sigma\hat{r})}{\beta(\sigma)}\gamma(1,\hat{\theta}) & (3.89) \\ &= n\hat{r}^{n-1}\gamma(1,\hat{\theta}) & (3.90) \\ &= \hat{r}^{n-1}\gamma^{(1,0)}(1,\hat{\theta}). & (3.91) \end{aligned}$$

This derivative in γ simply removes the integral in equation (3.81), so that the equation above can be rewritten as

$$\frac{m(\hat{r},\hat{\theta})}{\hat{r}} = \hat{r}^{n-1}\frac{m(1,\hat{\theta})}{1}. \tag{3.92}$$

Introducing $g(\hat{\theta}) = \hat{m}(1,\hat{\theta})$ completes the proof of the claim. □

Claim 3. *The operation exponent n must be a positive integer.*

Proof of Claim 3. According to Schwartz's Paley-Weiner theorem, the Fourier transform of a compactly supported distribution has continuous derivatives of all orders at every point. The origin is a location of particular interest, because almost all choices of g and n will lead to a discontinuity there.

First note that $g(\hat{\theta})$ cannot vanish everywhere. In this case, the aperture filter implied by claim 1 would be a Dirac delta pinhole. This corresponds to the no-residual optical flow case, which cannot reveal depth, and violates the integrability requirement on κ.

If $\Re(n)$ is negative, then $\hat{r}^n$ will go to complex infinity at the origin, and if n is not an integer or has an imaginary part, repeated application of the power rule shows that some derivative of $\hat{m}$ will have an exponent n' with $\Re(n') < 0$ and the same discontinuity will arise. Specifically, the j^{th}-order derivative in $\hat{r}$ of $\hat{m}$ is

$$\frac{d^j \hat{m}}{d\hat{r}^j} = \begin{cases} 0, & n \in \mathbb{Z}^+,\ j > n \text{ or } n = 0, \\ g(\hat{\theta}) \frac{n!}{(n-j)!} \hat{r}^{n-j}, & \text{else.} \end{cases} \tag{3.93}$$

As $\hat{r}$ approaches zero, this derivative approaches complex infinity for $\Re(n-j) < 0$ (unless n is a positive integer or zero), has an essential singularity when $n - j$ is imaginary, and otherwise goes to zero. So, there is a discontinuity in some order derivative in $\hat{r}$ at the origin unless n is a nonnegative integer. Strictly speaking, it is discontinuities in the derivatives in ω_x and ω_y that are forbidden, but these follow directly, e.g. by Faà di Bruno's formula. Thus, n must be a nonnegative integer.

When $n = 0$, $\hat{m} = g(\hat{\theta})$ must take a constant value g to avoid a discontinuity at the origin. Note that $\hat{k}$ under unit depth scaling ($\sigma = 1$) is exactly $\hat{\kappa}$, and consider the corresponding filter implied by claim 1:

$$\hat{\kappa} = f(\hat{\theta}) e^{-w(1) g \ln(\hat{r})} = f(\hat{\theta}) \hat{r}^{-w(1) g}. \tag{3.94}$$

Because κ is integrable, the Riemann-Lebesgue lemma states that $\hat{\kappa}$ must vanish as $\hat{r}$ approaches infinity, so $\Re(w(1)g)$ must be positive. However, this implies an infinite discontinuity at the origin which also violates integrability assumptions: integrability of κ implies uniform continuity of $\hat{\kappa}$. Thus, $n \neq 0$. □

Claim 4. *$\hat{m}$ is a homogeneous polynomial of degree n.*

Proof of Claim 4. The previous claims show that for some positive integer n,

$$\hat{\kappa}(\hat{r}, \hat{\theta}) = f(\hat{\theta})\, e^{-w(1) g(\hat{\theta}) \hat{r}^n}. \tag{3.95}$$

By the Riemann-Lebesgue lemma, integrability of κ implies $\hat{\kappa}$ vanishes at infinity, which requires $\Re(w(1)g(\hat{\theta})) > 0$ for $\hat{\theta} \in [-\pi, \pi]$. Then, $g(\hat{\theta}) \neq 0$ on $[-\pi, \pi]$ and in both ω_x and ω_y, $\hat{m}$ has a pole at (complex) infinity.

Schwartz's Paley-Wiener theorem states that the Fourier transform of a compactly supported distribution can be extended to an entire function, i.e. one that is complex differentiable everywhere in $\mathbb{C}^2$. Proofs of the previous claims have used the smoothness of $\hat{m}$ that this theorem implies over real values of ω_x and ω_y, but this is a much more restrictive condition, as described in section 3.3.1. In fact, the only entire functions with a pole at infinity are polynomials. Because $\hat{m}$ is degree n along any radial slice, it must also be homogeneous. □

Claim 5. *$f(\hat{\theta}) = a_0 \in \mathbb{R}^+$ and $n = 2$.*

Proof of Claim 5. The previous claims state that for some positive integer n, and constants $c_0, \ldots, c_n \in \mathbb{R}$

$$\hat{\kappa}(\hat{r}, \hat{\theta}) = f(\hat{\theta})\, e^{-w(1) \sum_j c_j \omega_x^{n-j} \omega_y^j}. \tag{3.96}$$

Integrability of κ implies that $\hat{\kappa}$ is uniformly continuous, so $f(\hat{\theta})$ must be a constant a_0 to avoid a discontinuity in $\hat{\kappa}$ at the origin, where the exponential term goes to one. It must be real by the conjugate symmetry of $\hat{\kappa}$ induced by reality of κ, and it cannot be negative or zero without causing κ to be so as well.

The Riemann-Lebesgue lemma implies that n is even, because $\hat{\kappa}$ must vanish as each ω approaches either positive or negative infinity while the sign of each c_j is fixed. For n even, each radial slice is an even function:

$$\begin{aligned}
g(\hat{\theta}) &= \sum_j c_j \cos^{n-j}(\hat{\theta}) \sin^j(\hat{\theta}) && (3.97)\\
&= \sum_j c_j (-1)^{n-j} \cos^{n-j}(\hat{\theta} + \pi)(-1)^j \sin^j(\hat{\theta} + \pi) && (3.98)\\
&= g(\hat{\theta} + \pi). && (3.99)
\end{aligned}$$

Next see that for $n \geq 3$, $\hat{\kappa}$ along any fixed-$\hat{\theta}$ radial slice is not a positive definite function, because with

$$C(n) = \sum\sum z_i z_j a_0 e^{-w(1)g(\hat{\theta})|r_i - r_j|^n}, \tag{3.100}$$

$$z = [1,\ -2,\ 1], \tag{3.101}$$

$$r = \left[-\sqrt[n]{\frac{.1}{w(1)g(\hat{\theta})}},\ 0,\ \sqrt[n]{\frac{.1}{w(1)g(\hat{\theta})}} \right], \tag{3.102}$$

it can be seen that

$$C(n) = 6 - 8e^{-.1} + 2e^{-(.1)(2^n)}, \tag{3.103}$$

and both $C(3)$ and $\frac{dC}{dn}$ are negative. The Fourier slice theorem [10, 81] states that each of these angular slices is the one-dimensional Fourier transform of the projection of κ along the same angle in the spatial domain:

$$\hat{\kappa}(\hat{r}, \theta_0) = \mathcal{F}_{1D}\left[\int \kappa(r\cos\theta_0 + z\sin\theta_0, r\sin\theta_0 - z\cos\theta_0)dz \right]. \tag{3.104}$$

However, Bochner's theorem states that the Fourier transform of a nonnegative integrable function must be positive definite. So for $n \geq 3$, all projections of κ, and therefore the filters κ themselves, fail to meet the requirements of nonnegativity and integrability. □

Proof of the Theorem. Combining the previous claims shows that

$$\hat{m} = c_0\omega_x^2 + c_1\omega_x\omega_y + c_2\omega_y^2 \tag{3.105}$$

$$\hat{\kappa} = a_0 e^{-w(1)(c_0\omega_x^2 + c_1\omega_x\omega_y + c_2\omega_y^2)}. \tag{3.106}$$

Integrability of κ requires that $w(1) > 0$ and $c_1^2 < 4c_0c_2$, so that for

$$\mathbf{\Sigma} = \begin{bmatrix} c_2 & -c_1/2 \\ -c_1/2 & c_0 \end{bmatrix} w(1), \tag{3.107}$$

the inverse Fourier transforms and basic manipulation prove the theorem. □

4
Prototypes

A pair of prototype sensors demonstrate the practicality of the depth from differential defocus cue. The first appears in [4, 5] and is a completely unactuated camera that observes a moving scene. It accumulates the defocus brightness constancy constraint over patches of at least four pixels and recovers the depth and 3D velocity at the back-projection of each patch. The second, seen in [39], augments the camera with an electronically-controlled deformable lens which oscillates its optical power over time. This sensor computes depth and confidence at every pixel without requiring scene or camera motion, and it dramatically extends its working range through tracking accommodation.

The previous chapter presented a theory that described a large class of depth sensors but that also depended on several physical and computational assumptions. Physically, it assumed that the world consists of a single plane that is parallel to the camera sensor, that the brightness of this world plane remains constant over time, and that the blur kernels of the camera are perfectly Gaussian and their size is perfectly determined by an aberration-free thin lens model. Computationally, it assumed that image derivatives would always be both accurate and useful, implying that the scene would have texture everywhere and that it could be measured with infinite spatial and temporal resolution, infinite bit depth, and no noise. Clearly, none of these assumptions are true. This chapter describes the approaches used to ensure both physical and computational robustness in moving from theory to practice. In doing so, it demonstrates the promise of the differential defocus cue for real-world sensing applications.

4.1 Per-Patch Depth and Velocity from an Unactuated Camera

The first prototype is targeted to extremely low power applications, such as microrobots and sensor nodes. These platforms cannot afford to expend power on camera actuation but may already experience motion in the scenes they need to measure, either by being attached to a moving chassis or by only needing to report changes in their inputs.

In the case of an unactuated camera observing a moving scene, the defocus brightness constancy constraint takes the form

$$I_t = \begin{bmatrix} I_x & I_y & (xI_x + yI_y) & (I_{xx} + I_{yy}) \end{bmatrix} \begin{bmatrix} \dot{X}f/Z \\ \dot{Y}f/Z \\ \dot{Z}/Z \\ \left(\frac{1}{Z_f} - \frac{1}{Z}\right) a\dot{Z}/Z \end{bmatrix}, \tag{4.1}$$

where Z and $(\dot{X}, \dot{Y}, \dot{Z})$ are the depth and velocity to be recovered, f is the fixed location of the photosensor relative to the lens and Z_f the fixed in-focus depth of the camera, and a is a calibrated constant determined by the fixed camera parameters. Rather than calibrating each camera parameter independently and combining them according to the physical model, f and a are calibrated directly to optimize depth-sensing performance on a calibration data set collected by moving textured planes on a translation stage. For details on the calibration procedure, see [5].

The algorithm implied by this constraint is simple: accumulate image derivatives in a patch of at least four pixels, take the least squares estimate of the four-vector of coefficients that is one-to-one with depth and 3D scene velocity, and recover the desired quantities using calibrated values for the parameters in equation 4.1.

This algorithm is applied to image pairs collected under small position changes using the camera shown in figure 4.1.1. This simple camera consists of a high-quality monochrome sensor (Grasshopper GS3-U3-23S6M-C from Point Grey Research), a single thin lens (LA1509-A from Thorlabs), and an optional apodizing filter (NDYR20B from Thorlabs). The filter is designed to turn the pillbox blur kernels of an ideal thin lens into Gaussians. From the measured blur kernels shown in figure 4.1.2, it is clear that the kernels deviate significantly from theory. Both with and without the apodizing filter, the blur kernels of the sensor are obviously non-Gaussian.

This indicates a possible problem for the system: the previous chapter proved that Gaussian blur is the only way to satisfy an analytically exact brightness constancy constraint. With non-Gaussian blur kernels, only an approximate constraint could hold. Happily, the results shown in this section demonstrate that the approximate constraint is sufficient to recover accurate depth and velocity, with or without the apodizing filter. The prototype shown in figure 4.1.1 measures depth within ±6mm over a range of 20cm using a one-inch-diameter lens. Figure 4.1.3 shows two examples of depth maps measured by this sensor. The image pairs used to compute these results violated physical assumptions on both the camera (with its non-Gaussian blur kernels) and the scene (with multiple and slanted planes), but the depth is still recovered accurately.

Computational robustness is achieved by the simple and well-established

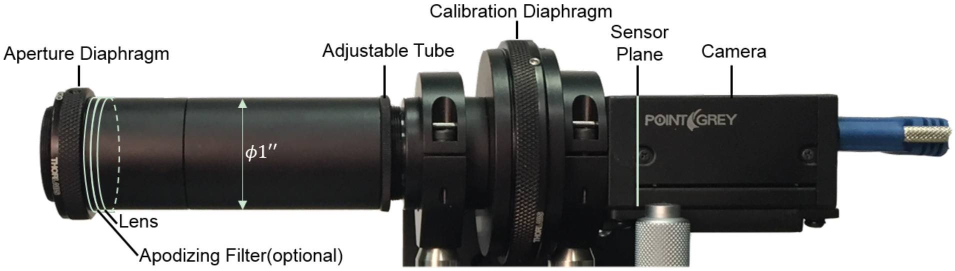


Figure 4.1.1: An unactuated camera prototype. A simple camera is used for depth and velocity detection on moving scenes. A monochrome photosensor observes the scene through a single 1”-diameter thin lens held at an adjustable location by a lens tube. Optionally, an apodizing filter can be placed at the aperture to make the blur kernels more Gaussian (see figure 4.1.2), but this has little effect on the sensor’s results. Figure courtesy of Qi Guo.

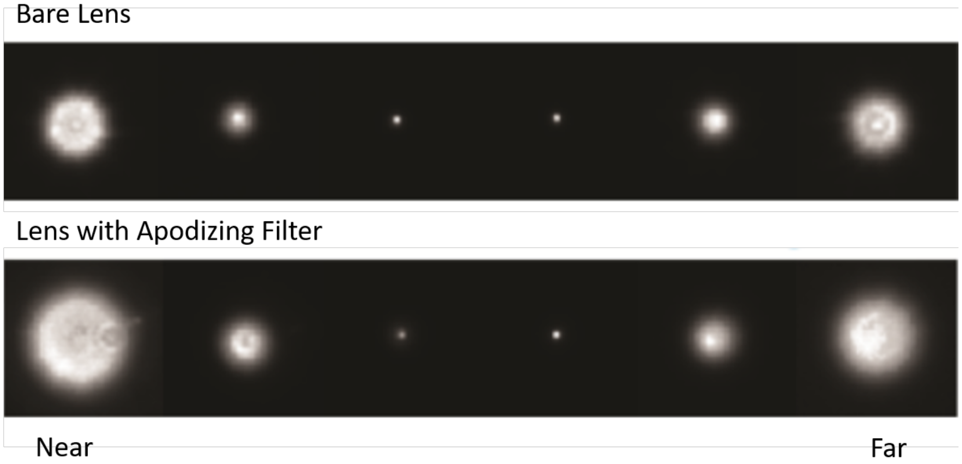


Figure 4.1.2: Blur kernels of the unactuated prototype. Blur kernels measured from the prototype camera shown in figure 4.1.1 from a central image patch as a point light source comes into and out of focus. They show obvious deviations from the Gaussian model, both with and without an apodizing filter. Figure 4.1.6 demonstrates that the two configurations provide similar performance. Figure courtesy of Qi Guo.

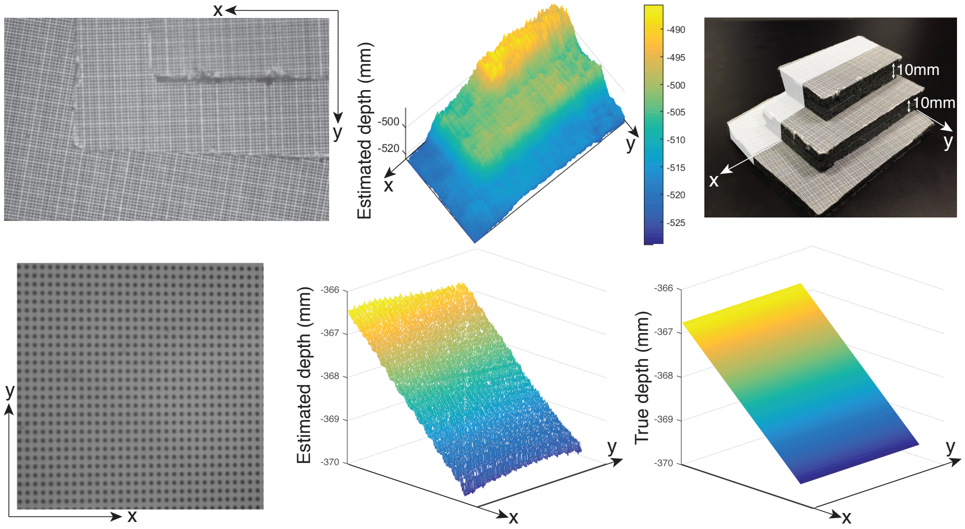


Figure 4.1.3: Depth recovered from simple scenes. From left to right: one frame from an input three-frame image sequence during which the object was moved on a translation stage; per-pixel depth measured by independent focal flow reconstruction in overlapping square windows; and true scene shape. The depth (and velocity) of these scenes is well-recovered despite the theoretical assumption of a single front-parallel plane. Figure courtesy of Qi Guo.

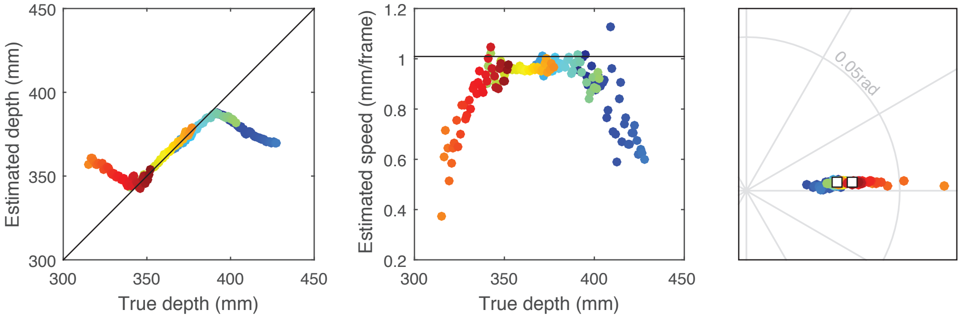


Figure 4.1.4: Working range for depth and velocity. Measured depth, speed, and 3D direction $(\dot{X}, \dot{Y}, \dot{Z})/\|(\dot{X}, \dot{Y}, \dot{Z})\|$ versus true depth, with markers colored by true depth. Directions shown by orthographic projection to XY-plane, where the view direction is the origin. Ground truth is black lines for depth and speed, and white squares for direction. Two ground truth directions result from remounting a translation stage to gain sufficient travel. Note that near the in-focus plane, indicated by yellow, orange, and teal markers, the depth and velocity are well-recovered, while away from the in-focus plane, indicated by red and blue markers, the recovered depth tends to the in-focus distance and the recovered velocity tends to zero. Figure courtesy of Todd Zickler.

method of increasing the patch size over which the image derivatives are accumulated. This approach has a long history in the optical flow literature [69]. An appropriate window size for this method can be chosen ahead of time from a model of the statistics of scenes that a given application is expected to encounter, or adjusted in response to image input. The window sizes for results shown in this section were selected by hand during post-processing.

Section 3.2.2 analyzed the breakdown of the differential defocus cue under large blur. This happens as per-pixel image differences, across neighboring pixels and within a single pixel over time, fall below the noise level. How quickly the cue breaks down depends on the amount of contrast available in the scene as well as the depth of field, time and space resolution, and bit depth of the camera. Figure 4.1.4 demonstrates high-quality recovery of both depth and velocity when objects are near the in-focus plane, and poor performance away from this region, tending toward the default solution of an in-focus and unmoving scene under large blur.

To quantify this behavior, let the working range for this prototype be defined as the range of depths for which the error in the depth measurement is within 1% of the in-focus depth Z_f. The size and quality of this working range can be adjusted by changing camera parameters, as shown in figure 4.1.5. Essentially, there is a fixed amount of signal available in the contrast of a given texture, and that signal can decay slowly over a large but less precise working range, or it can decay quickly to provide high quality measurements within a smaller working range. This tradeoff must be made at design time based on the anticipated application of the sensor.

Figures 4.1.5 and 4.1.6 show that, for this prototype, the working range is not particularly affected by the inclusion of an apodizing filter or a standard round aperture of the same approximate width. However, recall that the blur kernels in figure 4.1.2 show that both optical configurations produce clear deviations from the desired Gaussian blur kernels.

This prototype demonstrates that a standard, unactuated wide aperture camera can be used to detect the depth and 3D velocity of a moving scene accurately and efficiently within its working range, in spite of physical and computational nonidealities.

4.2 Per-Pixel Depth and Confidence from an Accommodating Lens

Two major weaknesses of the previous prototype are its blindness to static scenes and its fixed working range. Both of these issues can be addressed with a small additional expenditure of electrical power. This section describes the results of augmenting the previous sensor with an electronically-controlled deformable lens. This lens, which is a commercially available component, can be reshaped by applying an electric current, allowing dynamic control of the optical power of the lens.

The prototype, shown in figure 4.2.1, consists of a monochrome camera (Point Grey GS3-U3-23S6M-C) and a deformable lens (Optotune

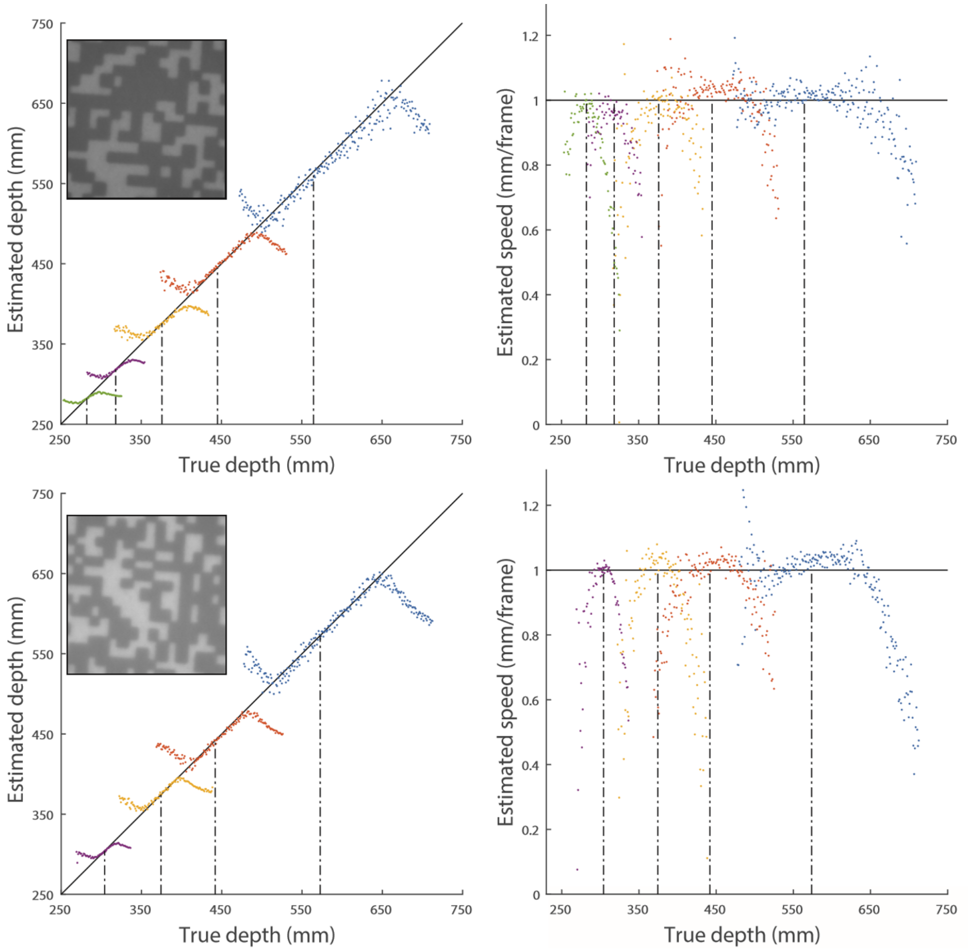


Figure 4.1.5: Working range for several lens locations. Adjusting the distance between the lens and the photosensor changes the in-focus depth, which moves and scales the working range. This indicates how the working range can be set at design time as described in section 3.2.2. Shown here are depth and velocity measurements for several lens locations (indicated by color with in-focus depth shown with dotted lines) with ground truth on solid lines. Inset are sample images from the data used to evaluate the system, which consisted of ten shots averaged to reduce noise. Top row shows results from a bare lens and the bottom row shows results from images collected through an apodizing filter. Figure courtesy of Qi Guo.

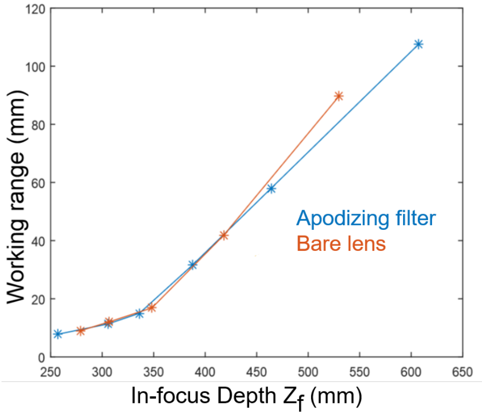


Figure 4.1.6: Performance with and without apodizing fiter. The working ranges for several lens locations are shown. In blue, the lens was augmented with an apodizing filter, designed to transform pillbox kernels to Gaussians. In orange, the apodizing filter was replaced with a standard round aperture. Figure 4.1.2 shows that the point spread functions in both situations are clearly non-Gaussian, and this data demonstrates that the difference between the two sets of kernels does not have an appreciable effect on sensor performance.

EL-10-30-C-VIS-LD-MV). This deformable lens has also been used for focal sweep imaging [74] and multifocal displays [22]. Additionally, a traditional thin lens provides a constant offset to the focal plane of the deformable lens. Similar to the effects seen in the previous prototype, this fixed lens provides a degree of freedom that allows a tradeoff between a large, low-resolution working range, or a smaller, higher-quality one. The effects of two different offset lenses are demonstrated later in this chapter.

The lens and camera are connected with control electronics that oscillate the optical power of the lens with an adjustable midpoint, amplitude, and frequency, and the peak and trough of this oscillation are synced with image capture. The plot in the top left of figure 4.2.1 indicates the electrical signal sent to both the lens and the shutter. It is important to note that the capture time is small with respect to the oscillation time, so the images received are differentially defocused image pairs (with $\Delta p = \pm .4$ diopters) rather than focal sweep images. Also note that this oscillation allows the optical power of the lens to be driven much faster than it would be if independent values were requested for each frame, due to the mechanical settling time of the lens. It is capable of normal-mode oscillations up to about 100 Hz. This places an upper bound on the frame rate of a focal track system, but it is likely to increase in the future as deformable lens technology continues to evolve (e.g. [99]). The blur kernels generated by this liquid lens are almost pathologically non-Gaussian, as shown in figure 4.2.2, but the defocus brightness constancy constraint still provides useful depth information.

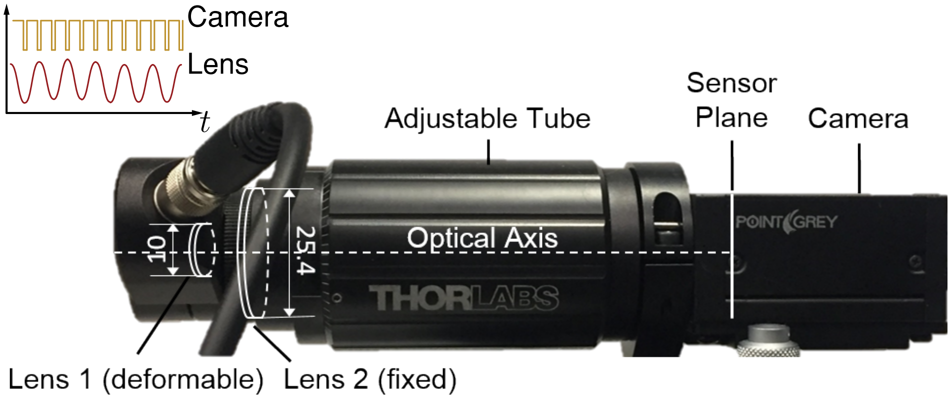


Figure 4.2.1: A camera prototype with electronically-driven optical power. A monochrome camera observes the world through a fixed lens and a deformable lens. Custom electronics oscillate the optical power of the lens and synchronize image capture to the peak and trough of each oscillation (plot in upper left) so that images are captured in differentially-defocused pairs. The centerpoint of this oscillation is determined by both the fixed lens and the tracking accommodation of the deformable lens illustrated in figures 4.2.6 and 4.2.7. Figure courtesy of Qi Guo.

4.2.1 Computing Depth and Confidence Maps

In the case of a static scene observed through a radially-symmetric deforming lens, the defocus brightness constancy constraint takes the form

$$I_t = \begin{bmatrix} I_x & I_y & (xI_x + yI_y) & (I_{xx} + I_{yy}) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \left(\frac{1}{Z_f} - \frac{1}{Z}\right) \beta \dot{p} \end{bmatrix}, \qquad (4.2)$$

where, again, Z is the depth to be recovered and Z_f the in-focus depth (that is now varying over time, but assumed to be known), with β a fixed and calibrated camera parameter and $\dot{p}$ the rate of change of the optical power. Rather than suggesting a patchwise matrix measurement of four scene parameters, this

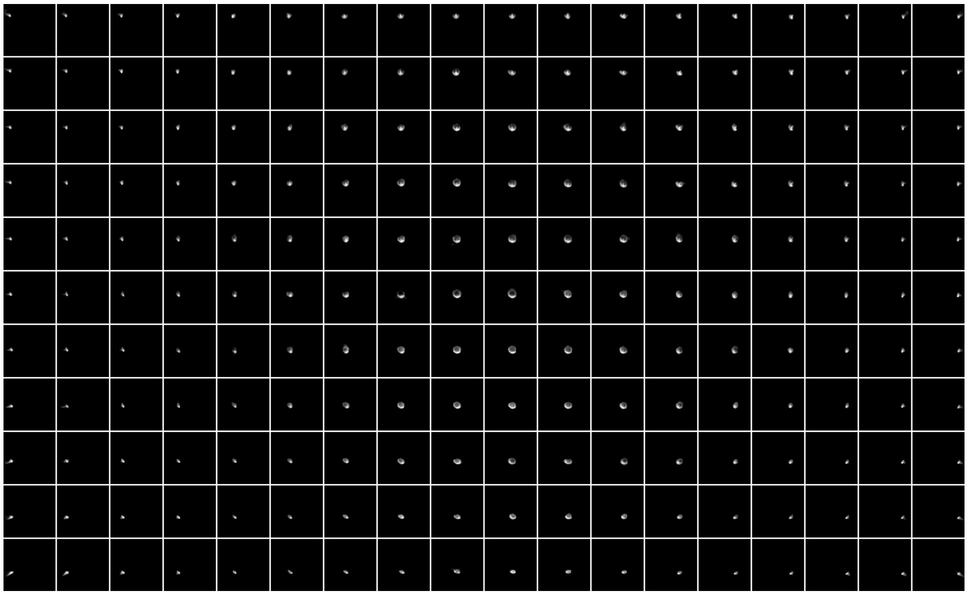


Figure 4.2.2: Blur kernels of the actuated prototype. Blur kernels of the actuated lens demonstrate high spatial variation and dramatic deviations from the Gaussian model. Shown here are the kernels measured for a single depth across the image plane. They display a great degree of radial distortion, bottom-heaviness likely due to the effects of gravity, and higher order effects possibly due to the mechanical oscillation of the lens surface. Figure courtesy of Dor Verbin.

constraint immediately implies a per-pixel equation for depth:

$$Z = \frac{\beta \dot{p} \nabla^2 I}{\frac{\beta \dot{p}}{Z_f} \nabla^2 I - I_t} = \frac{\beta \dot{p} \nabla^2 I}{\frac{fp-1}{f} \beta \dot{p} \nabla^2 I - I_t}. \tag{4.3}$$

As described in the sensitiyivty analysis of section 3.2.2, the defocus brightness constancy constraint holds true at every pixel in an image, but it will not always be useful. In a given natural image, there will be regions where the described method fails. These include regions showing textureless patches, objects that are severely out of focus, points near depth discontinuities, and so on. Any depth from defocus method, stereo approach, or other triangulation-based algorithm suffers from this weakness. Fortunately, image content indicates where these difficult regions are likely to be. This can be used to augment each depth measurement with an estimate of the appropriate confidence to have in that measurement. These general-purpose confidence maps can be used in many ways, from discarding low-confidence estimates to fusing results from different cues or computations.

There are many ways to estimate this confidence. Simple metrics include basic estimates of the amount of image contrast in the neighborhood, such as $|I_x|$ or $|\nabla^2 I|$, or a more sophisticated estimate like the focus measures reviewed in [87]. More complex approaches could include training a general neural network to predict confidence values from the image. It is not yet clear how performance and computational expense trade off among different confidence estimation architectures.

The confidence metric used in the results shown here is based on the observation that the depth equation can be written as a ratio,

$$Z = \frac{V}{W}, \tag{4.4}$$

$$V = \beta \dot{p} \nabla^2 I, \tag{4.5}$$

$$W = \frac{\beta \dot{p}}{Z_f} \nabla^2 I - I_t. \tag{4.6}$$

To first order, the variance of the measurement Z from the ratio V/W will be

$$\mathrm{Var}[\tilde{Z}] \approx \frac{\mathrm{E}[\tilde{V}]^2}{\mathrm{E}[\tilde{W}]^2}\left(\frac{\mathrm{Var}[\tilde{V}]}{\mathrm{E}[\tilde{V}]^2} + \frac{\mathrm{Var}[\tilde{W}]}{\mathrm{E}[\tilde{W}]^2} - \frac{2\mathrm{Cov}[\tilde{V}, \tilde{W}]}{\mathrm{E}[\tilde{V}]\mathrm{E}[\tilde{W}]}\right). \tag{4.7}$$

Estimating these expectations and variances would require statistical measurement over spatiotemporal patches, but experiments below indicate that reasonable results can be obtained by deriving expressions for $\mathrm{Var}[\tilde{V}]$, $\mathrm{Var}[\tilde{W}]$, and $\mathrm{Cov}[\tilde{V}, \tilde{W}]$ from an additive zero-mean Gaussian image noise model, and using single measurements of V and W as proxies for $\mathrm{E}[\tilde{V}]$ and $\mathrm{E}[\tilde{W}]$. Performance would almost certainly improve from properly estimating the expectation values.

Confidence should increase with a decrease in measurement variance. Scaling the inverse of the estimated variance of $\tilde{Z}$ to the range $[0,1]$, gives a parametric confidence model with three tuneable parameters, $\omega_0, \omega_1, \omega_2$:

$$C = \left(\frac{\omega_0}{W^2} + \frac{V^2\omega_1}{W^4} + \frac{V\omega_2}{W^3} + 1\right)^{-1/2}, \tag{4.8}$$

which are determined by the noise level, camera parameters, and computation details. The depth parameter β and confidence parameters $\vec{\omega}$ are calibrated simultaneously on translation stage image data using backpropagation as described in [39].

Figure 4.2.3 shows the result of applying these two equations at every pixel in a differentially defocused image pair collected by the sensor. It is apparent that the correct depth (in red) is only measured in regions with strong texture, while in texture-free patches the depth equation fails and low values in the confidence map reflect this poor performance. This depth and confidence map can be used as the input to a general processing pipeline, for example one that densifies the depth maps based on confidence and the collected images as in [72]. This work focuses instead on refining the depth and confidence map through robust computation, and the results reported here are simply depth maps thresholded to only include pixels with high confidence.

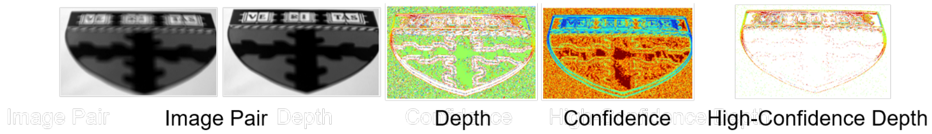


Figure 4.2.3: An image pair provides per-pixel depth and confidence. The image pair on the left is collected by the prototype under a small change in optical power observing a slanted textured plane. Note that in the first image, the bottom of the shield is blurrier than in the second. Applying equations 4.3 and 4.8 gives depth and confidence at each pixel. Near texture edges, the correct depth (red) is recovered, while in texture-free regions the default solution Z_f (green) is returned. Correct depth estimates correspond to high confidence scores (blue) and incorrect ones to low confidence (red), but these accurate pixels are sparse, illustrated by the confidence-thresholded output on the far right.

4.2.2 Computational Robustness

Two main approaches increase the robustness of this sensor's depth computation. The first is a standard multiscale image pyramid, where the input image pair is downsampled several times and per-pixel depth and confidence are computed at every scale. The other data augmentation technique is the use of multiple derivative orders. This comes from the observation that the constraint,

$$I_t = \left(\frac{1}{Z_f} - \frac{1}{Z} \right) \beta \dot{p} \nabla^2 I, \tag{4.9}$$

is spatially invariant, so that

$$\begin{aligned} \partial_x(I_t) =& \partial_x \left(\left(\frac{1}{Z_f} - \frac{1}{Z} \right) \beta \dot{p} \nabla^2 I \right) = \left(\frac{1}{Z_f} - \frac{1}{Z} \right) \beta \dot{p}\, \partial_x(\nabla^2 I), && (4.10) \\ \partial_y(I_t) =& \left(\frac{1}{Z_f} - \frac{1}{Z} \right) \beta \dot{p}\, \partial_y(\nabla^2 I), && (4.11) \\ \partial_{xy}(I_t) =& \left(\frac{1}{Z_f} - \frac{1}{Z} \right) \beta \dot{p}\, \partial_{xy}(\nabla^2 I), && (4.12) \end{aligned}$$

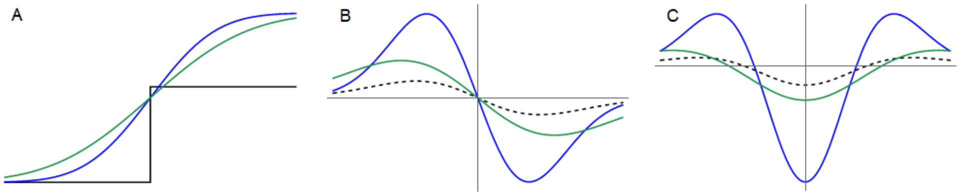


Figure 4.2.4: An illustration of texture-dependent confidence. A. A texture edge, in black, is viewed under two different blur conditions to generate the green and blue images. **B.** The per-pixel difference in these images I_t, shown in dashed black, vanishes at the center of the edge, as do the image Laplacians in blue and green. Depth equation based on $\nabla^2 I/I_t$ becomes unstable near this point. **C.** The spatial derivatives of each of the quantities in **B** are nonzero at the edge center and can provide useful depth information from the quantity $\partial_x(\nabla^2 I)/\partial_x(I_t)$. A combination of these derivative orders based on local texture content provides robust depth measurements.

and so on. The relationship between, for example, I_{tx} and $I_{xxx} + I_{yyx}$ rather than I_t and $I_{xx} + I_{yy}$ provides high-confidence results at different image locations because their values will have zero crossings at different locations across a given texture feature. Though higher-order derivatives are more sensitive to image noise, in the breakdown locations for lower-order derivatives they may still produce reliable measurements. Figure 4.2.4 shows the effect of Gaussian blur of changing scale on an image edge. Notice that at the center of the edge, both the Laplacian and the image difference go to zero, leading to an instability in the signal $\nabla^2 I/I_t$. The breakdown shifts spatially for $\partial_x(\nabla^2 I)/\ \partial_x(I_t)$. This indicates that a confidence-weighted combination of depth and confidence measurements from multiple image scales and derivative orders could provide denser and higher quality depth maps.

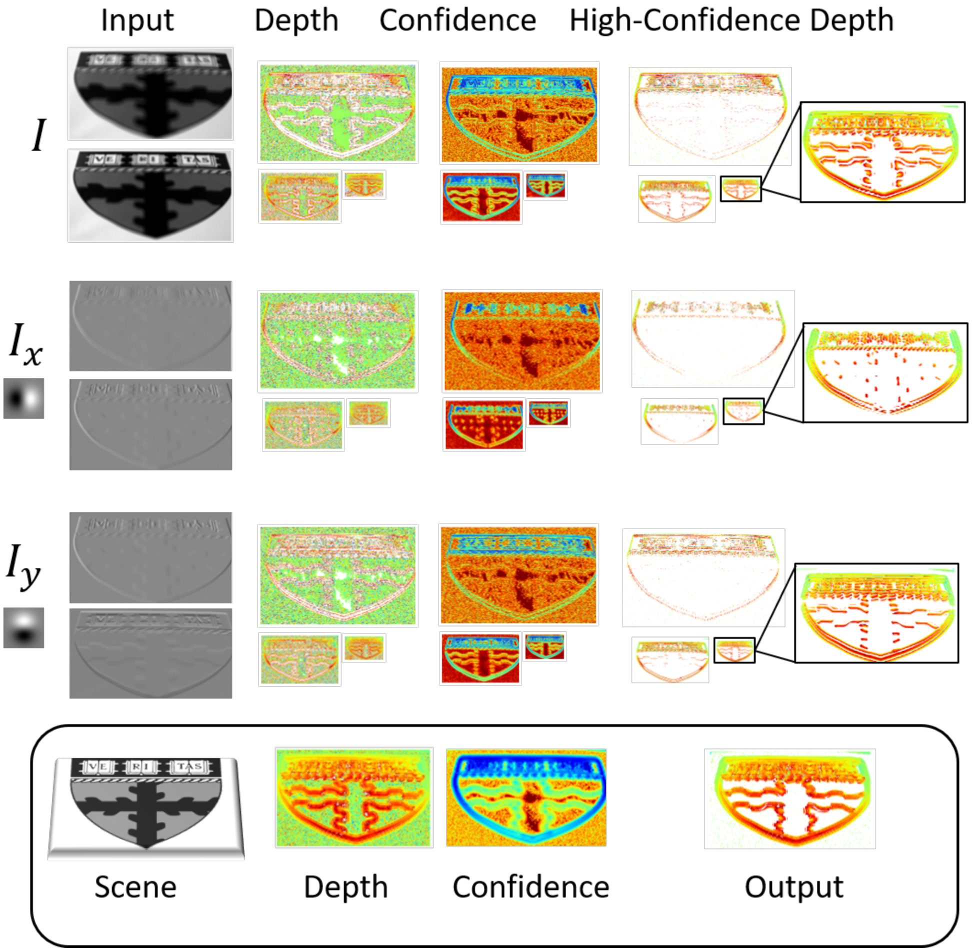
The individual depth and confidence maps produce the final output when combined with softmax:

$$\begin{aligned}(Z, C) &= \sum_{i,j} \gamma^{i,j}(Z^{i,j}, C^{i,j}),\\ \gamma^{i,j} &= \frac{\exp(C^{i,j})}{\sum\limits_{i,j} \exp(C^{i,j})},\end{aligned} \tag{4.13}$$

where superscript i indicates the image scale and j the derivative order. An example of confidence-thresholded depth maps computed from several image scales and derivative orders is shown in figure 4.2.5. Note the insets on the right that illustrate the texture-dependence of confidence and depth accuracy for different image feature orientations, and the reduced texture dependency and increased high-confidence density of the final depth map.

The robust computational pyramid illustrated in figure 4.2.5 provides dense and accurate depth measurements in textured image regions, and it does so

Figure 4.2.5 *(following page)*: Robust computation through multiple scales and derivative orders. To generate denser and higher-quality output, depth and confidence are computed at multiple scales and derivative orders. The first row, using the original image pair, shows the same depth and confidence computations as in figure 4.2.3 performed on a three-scale pyramid. High confidence depth measurements are still sparse, and the inset on the right shows low confidence at edge centers, as expected from the principle illustrated in figure 4.2.4. The second row shows the results from applying equations 4.3 and 4.8 to the x derivatives of each of the collected images. High-confidence pixels now appear in the center of vertical edges. Likewise, the third row shows the result of computations on the y derivative of the input images, which gives good results at the center of horizontal edges. Combining these scales and derivative orders provides robustness to scene texture: the box at the bottom shows a schematic of the tilted plane scene, the depth and confidence map created by combining the pyramid of estimates with softmax according to equation 4.13, and the final output, which shows accurate measurements in red near texture edges and does not report depth in low-confidence textureless regions. This computation includes eighteen parameters trained with backpropagation and returns 300 × 480 pixel results at 100 frames per second on a laptop GPU.


Input
Depth
Confidence
High-Confidence Depth
I
I_x
I_y
VE RI TAS
Scene
Depth
Confidence
Output

without sacrificing the computational efficiency enabled by the differential defocus approach. In practice, this prototype uses three image scales (1, .5, & .25) and three derivative orders (I, I_x, & I_y), with a total of eighteen calibrated depth and confidence parameters, and is able to output full-resolution (300 × 480) depth and confidence maps at 100 frames per second on a laptop GPU.

4.2.3 Physical Robustness

Robust computation architectures do not address the fundamental loss of contrast away from the in-focus plane, and the sensor still has a finite working range. However, because the location of the in-focus plane of this sensor is under active control, the working range can be significantly increased by adjusting the center point of the focal plane oscillation to match an object of interest in the scene. Specifically, the PID controller shown in figure 4.2.6 slowly matches the in-focus plane of the lens to the median of high-confidence depth measurements while continuing to oscillate the lens power. The advantage of this large-scale accommodation for depth accuracy can be observed in figure 4.2.7, which shows an object approaching the camera with and without accommodation.

The working range can also be adjusted by changing the fixed offset lens. Figures 4.2.7 and 4.2.8 show results using a 10-diopter lens (Thorlabs LA1509-A), well suited to recover fine details on small objects like children's toys, with figure 4.2.8 showing the effect of changes in the confidence threshold. In this optical configuration, the prototype had a 75 cm range where the mean depth error was below 4 cm for a confidence threshold of 0.995. Figure 4.2.9 shows results for a larger field of view, accomplished by using a 25-diopter lens (Thorlabs AC254-040-A-ML) located closer to the sensor. In this configuration, which is better suited to larger objects such as a gesturing hand, the prototype had a 1 m range where the mean depth error was below 5 cm for a confidence threshold of 0.995.

These results show that even with the extremely non-Gaussian blur kernels shown in figure 4.2.2, the prototype can measure depth accurately, densely, and

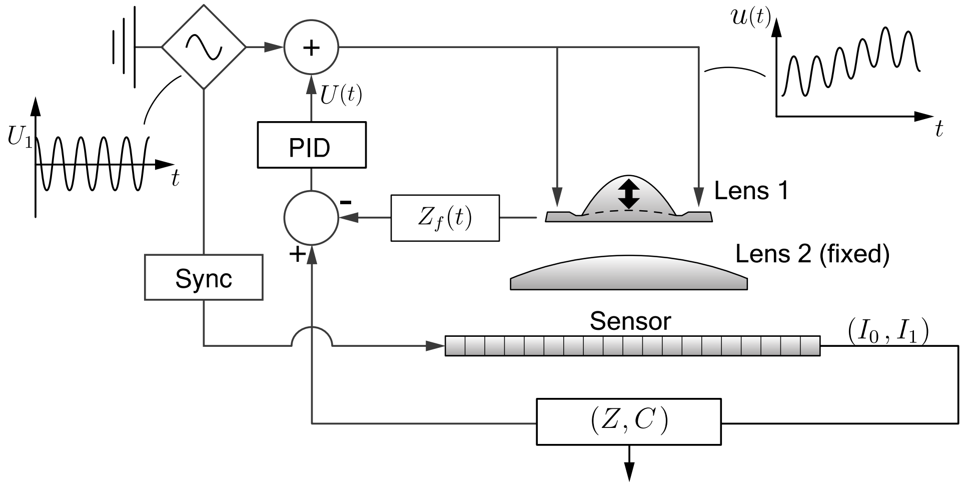


Figure 4.2.6: Feedback control for accommodation. The median of the confidently-measured scene depth feeds back to the lens controller, as a slow-changing offset to the sinusoidal control signal. This keeps the scene in focus, where the defocus cue is strongest, for an extended working range. Figure courtesy of Qi Guo.

efficiently in textured regions with a combination of augmented computation and physical accommodation. This approach is sufficiently fast and robust that several of the underlying physical assumptions can be broken.

First, while the depth recovery equation assumes that there is no camera or scene motion, the high speed of computation means that in practice moving objects are measured accurately in spite of small frame-to-frame motion. On the human scale, it takes effort to move an object quickly enough that noticeable errors in depth measurement occur.

Second, far exceeding the assumption of a single front-parallel constant-brightness plane, the sensor performs well on highly optically challenging scenes, including shiny, transparent, and reflective objects with complicated and discontinuous shapes. Figure 4.2.9 demonstrates the ability of the sensor to greatly exceed expectations on objects such as bubble wrap, transparent plastic cups, CDs, and plastic crystals. Note that the edges of reflective facets are measured as on-object texture, while textured objects seen in

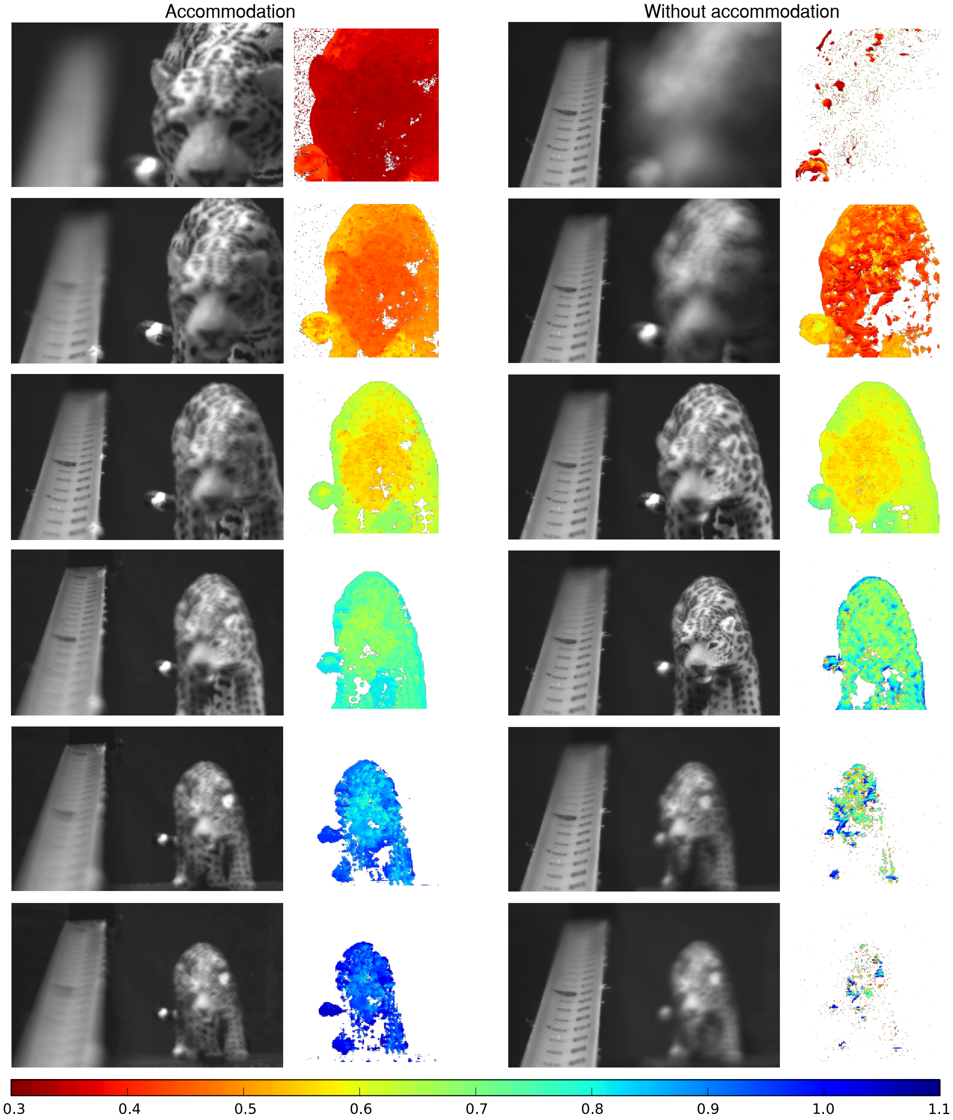


Figure 4.2.7: Working range extension through accommodation. Demonstration with (left) and without (right) accommodation for confidence > 0.995 and depth in meters. The accommodation controller uses a region of interest that contains the toy but not the tape measure, which is included to visualize the adapting focal depth. Accommodation more than doubles the useful range of the sensor. Figure courtesy of Qi Guo.

reflections result in the optical depth of the object, shown in figure 4.2.9.B. Also note the recovery of multiple scene planes through a transparent object in figure 4.2.9.C, which provides a significant argument against the method of immediately and universally densifying depth maps rather than directly reporting the per-pixel depth and confidence measurement. These examples demonstrate that for this passive, monocular depth cue, optical effects such as the edges of specularities, facets, caustics, and shadows, rather than leading to measurement errors, can be treated as valuable sources of additional texture on the object.

Figure 4.2.8 ***(following page)*****: Depth and confidence of everyday objects.** Small objects such as toys and slanted planes were observed by the sensor. Raising the confidence threshold makes the depth map sparser and more accurate. Depth shown in meters. Figure courtesy of Qi Guo.

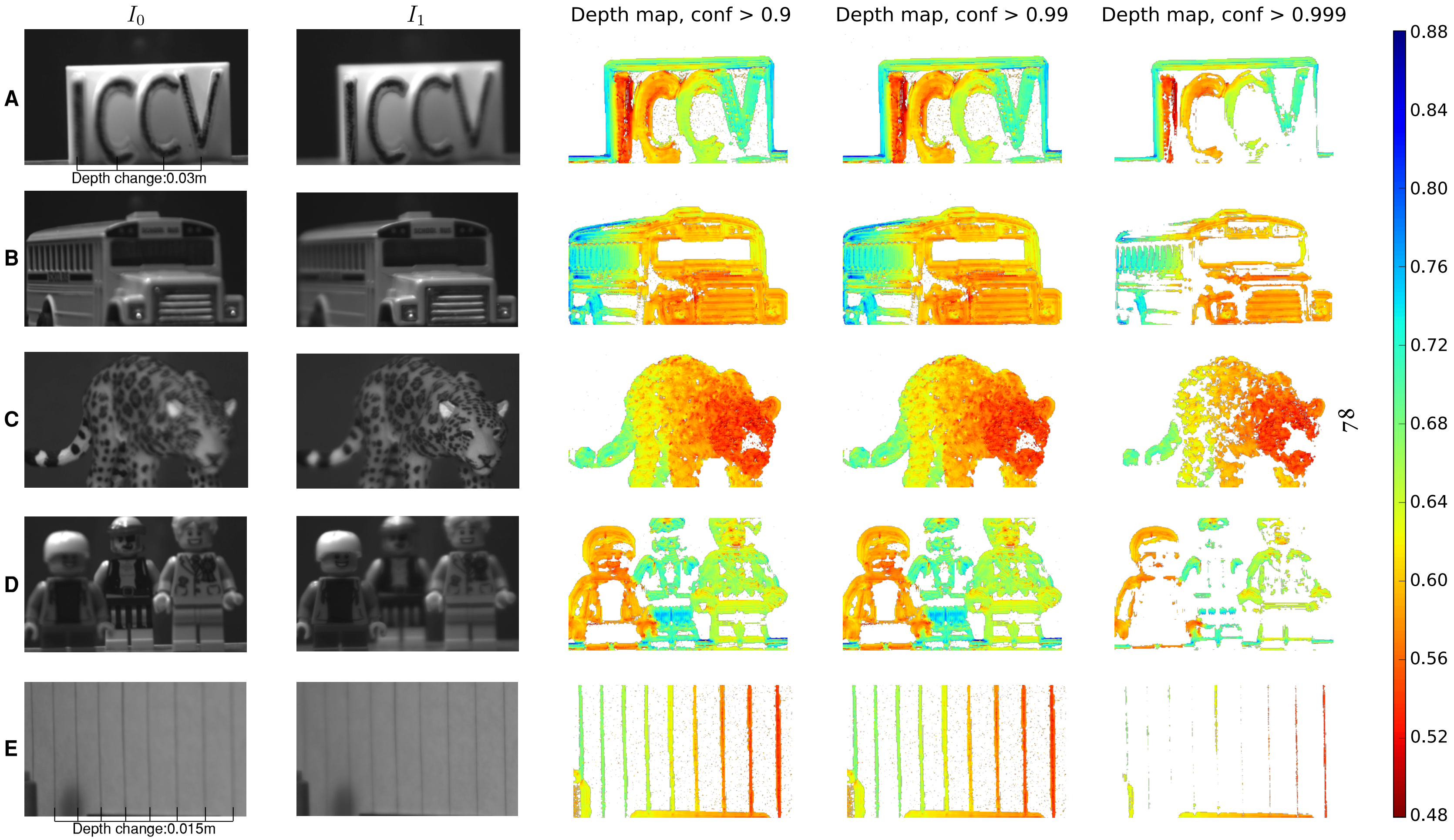
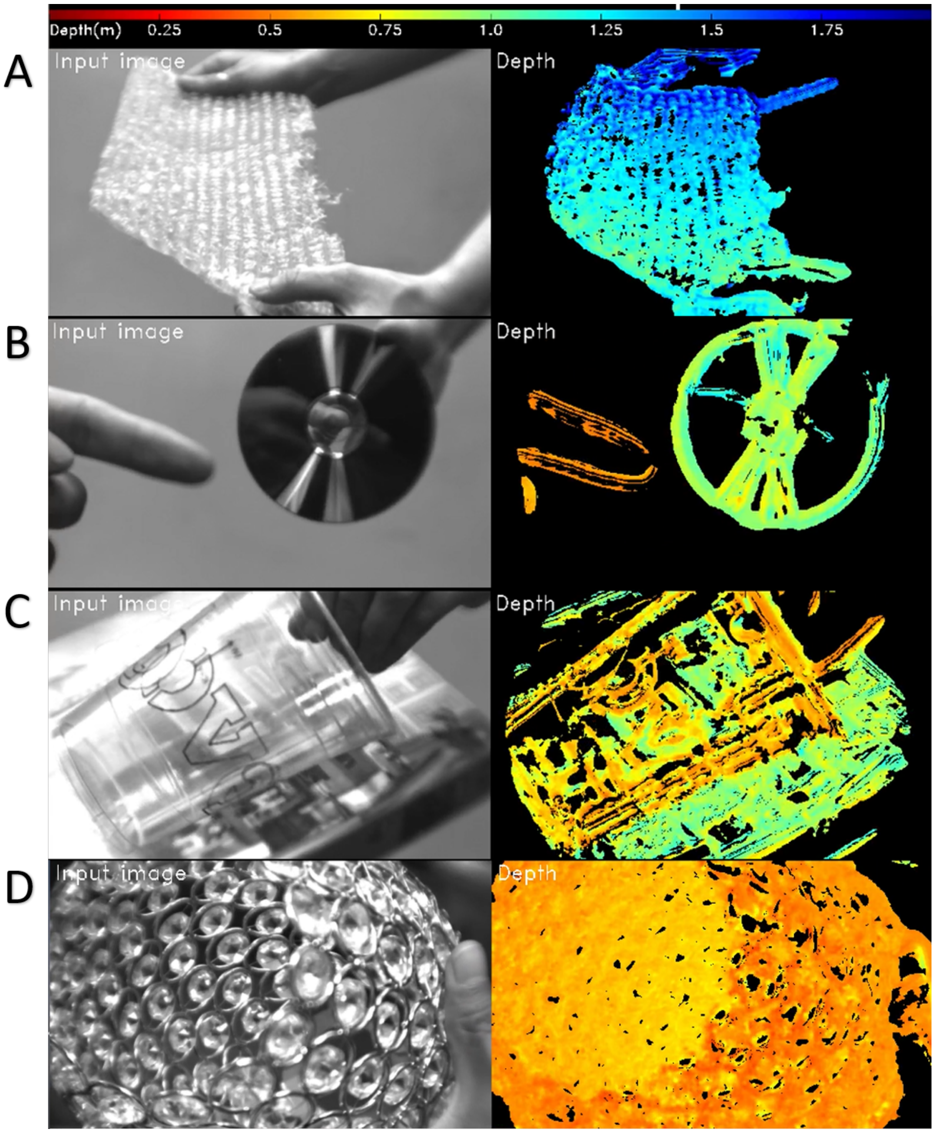

I_0
I_1
Depth map, conf > 0.9
Depth map, conf > 0.99
Depth map, conf > 0.999
A
B
C
D
E
Depth change:0.03m
Depth change:0.015m
0.88
0.84
0.80
0.76
0.72
0.68
0.64
0.60
0.56
0.52
0.48

Figure 4.2.9 ***(following page)*****: Depth measurement in optically-challenging scenes.** While the theory in chapter 3 assumed the camera would view a single front-parallel constant-brightness plane, the passive and monocular nature of depth from defocus allows this cue to apply to a number of optically challenging objects. Optical effects like reflections, caustics, and bokeh can be treated as additional sources of on-object texture and actually improve results. The sensor successfully recovers the depth of: **A** bubble wrap, which is a flexible, reflective, transparent, unmarked surface; **B** a CD showing both highlights from small-facet reflections of scene light, which register as on-object texture and provide the depth of the CD, as well as a large-facet reflection of the finger in the foreground, which appears as a more-distant line within the disc due to the increased optical distance of the reflected light; **C** the highlights and markings on a transparent cup simultaneously with the textured plane behind it, which illustrates a key advantage to the approach of reporting sparse confident depth maps instead of attempting to densify them; and **D** a highly discontinuous, shiny clear object made of crystals held together in a wire frame, which violates all of the physical assumptions of the theory but is nonetheless well-reconstructed by the sensor.


Depth(m)
0.25
0.5
0.75
1.0
1.25
1.5
1.75
A
Input image
Depth
B
Input image
Depth
C
Input image
Depth
D
Input image
Depth

5

Open Questions

While many theoretical and practical aspects of differential defocus have been explored in this document, many more questions remain.

Perhaps the most obvious area for exploration is suggested by the fact that only two prototypes have been presented, from a wide class of sensors that was described theoretically. One of these that seems to hold particular promise is the combination of camera and photosensor motion, which can be used to describe the moving lens assemblies common in cell phone cameras.

There are also many ways to depart from a simple thin lens camera to optically collect single-shot defocus pairs. One could imagine a photosensor with a pixel mosaicking of microlens optical powers, physical pixel depths, or polarization filters accompanied by a birefringent lens. Promising developments in optical component manufacturing suggest that diffractive lenses can be designed to generate custom blur kernels for different depths, colors, and polarizations. These lenses can be designed in close cooperation with the post-processing pipeline to maximize sensor performance.

To make any of these sensors as effective as they could be, several basic engineering questions would have to be answered. A major area for improvement in the current prototypes is the slapdash approach to confidence. Good confidence estimation is key to sensor performance, and a solid characterization of model- and learning-based confidence architectures could provide a major performance boost. This would likely be even more important for computational pipelines that densify the output depth maps to include estimates over texture-free regions, which this work does not consider.

Another major issue not addressed in this work is its heavy reliance on calibrated parameters. The prototypes shown here were assembled using high quality equipment in a controlled environment and then were calibrated by collecting data sets with known depth. Some relief comes from the fact that parameters were combined into a small number of needed weights, so that the full set of parameters was never measured, and that these values were optimized directly for depth-sensing performance rather than some intermediate camera model. Still, the calibration procedure was a major effort and highly specific to each single prototype. For devices that are manufactured with variance in the camera parameters, or for those that experience change in these parameters over the course of their use, the lengthy calibration pipelines used in this work would be completely infeasible. However, just as a moving sensor can recover velocity as well as depth, it is possible to recover limited camera information efficiently from image derivatives simultaneously with depth. Because many parameters of interest appear in conjunction with their derivatives, this limited parameter recovery combined with principled use of temporal filtering might offer a practical solution.

Additionally, while both the theory and the prototypes attempt to describe the effect of camera parameters on sensor performance, a great deal more data and analysis is needed to be able to answer practical questions about camera design. It is not even clear what the optimal blur change should be between image pairs, which many depth from defocus methods characterize. The ease of finding camera settings that produced good results discouraged this line of research in the proof-of-concept stage.

The enormous amount of implementation work that has not yet been done only highlights the mysterious robustness of the depth from differential defocus cue. In spite of strict theoretical assumptions — an ideal thin lens with perfectly Gaussian kernels observing a single front-parallel, constant-brightness plane — the prototypes were able to successfully recover depth through dramatically non-Gaussian kernels for optically challenging objects with complex and highly discontinuous geometries, such as bubble wrap and a wire mesh hung with plastic crystals. Why does the defocus brightness constancy constraint, when applied to images with non-Gaussian blur and with no attempt to account for this discrepancy, perform so well? Why does a blur model based on a single front-parallel constant-brightness plane apply to objects with complicated shapes, specularities, interreflections, and caustics? There is currently no explanation for why this should be possible.

Particularly interesting in this regard are reflective objects. For large shiny objects like a CD or an etched silver plate, on-mirror texture and highlights produce the depth of the reflective surface, while reflections of textured objects give the optical depth to the reflected object. For shiny objects like bubble wrap or plastic crystals, no clear reflection is formed, and all reflected light contributes to detection of the surface. It seems that this is related to reflective facet size, where facet edges provide the depth of the reflector and facet surfaces provide the depth of the reflected object. This observation suggests one potential way to characterize the dependence of the differential defocus cue on material parameters, which might provide far richer insight than the current constant brightness assumption.

One key optical phenomenon that was completely ignored in this work is the effect of wavelength. Chromatic aberration is known to assist human depth perception, and computer vision techniques to recover depth from chromatic aberration are plagued by the very small change in defocus that it generates, which may be an advantage in the differential defocus framework. The ability of colorblind cephalopods to successfully match the colors of scenes observed through strangely-shaped pupils suggests that both depth and color may be recoverable from monochromatic defocus information with appropriate aperture design, perhaps differentially.

It also remains to be seen how depth from differential defocus sensors can be optimized for the size and power requirements of the microscale sensing platforms that inspired them. Many engineering challenges remain in shrinking and streamlining the optics, electronics, and software that would be required. However, the efficiency and robustness of the depth sensors shown here, as well as the use of small defocus changes by the jumping spider, suggest that differential defocus may create possibilities for depth sensing on highly limited platforms.

References

[1] Yair Adato, Yuriy Vasilyev, Todd Zickler, and Ohad Ben-Shahar. Shape from specular flow. *Pattern Analysis and Machine Intelligence*, 32(11): 2054–2070, 2010.

[2] Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen. Interactive digital photomontage. *ACM Transactions on Graphics (ToG)*, 2004.

[3] Muhammad Bilal Ahmad and Tae Sun Choi. Application of three dimensional shape from image focus in lcd/tft displays manufacturing. *IEEE Transactions on Consumer Electronics*, 53(1), 2007.

[4] Emma Alexander, Qi Guo, Steven J Gortler, and Todd Zickler. Focal flow: Measuring distance and velocity with defocus and differential motion. In *European Conference on Computer Vision (ECCV)*, 2016.

[5] Emma Alexander, Qi Guo, Sanjeev Koppal, Steven J Gortler, and Todd Zickler. Focal flow: Velocity and depth from differential defocus through motion. *International Journal of Computer Vision*, pages 1–22, 2017.

[6] Youngeun An, Gwangwon Kang, Il-Jung Kim, Hyun-Sook Chung, and Jongan Park. Shape from focus through laplacian using 3d window. In *International Conference on Future Generation Communication and Networking*, 2008.

[7] J Baina and J Dublet. Automatic focus and iris control for video cameras. In *International Conference on Image Processing and its Applications*, 1995.

[8] Martin S Banks, William W Sprague, Jürgen Schmoll, Jared AQ Parnell, and Gordon D Love. Why do animal eyes have pupils of different shapes? *Science Advances*, 1(7):e1500391, 2015.

[9] AD Blest, RC Hardie, P McIntyre, and DS Williams. The spectral sensitivities of identified receptors and the function of retinal tiering in the principal eyes of a jumping spider. *Journal of Comparative Physiology*, 145(2):227–239, 1981.

[10] Ronald N Bracewell. Strip integration in radio astronomy. *Australian Journal of Physics*, 9(2):198–217, 1956.

[11] Michael J Brooks and Berthold KP Horn. Shape and source from shading. *International Joint Conferences on Artificial Intelligence (IJCAI)*, 1985.

[12] Anna R Bruss and Berthold KP Horn. Passive navigation. *Computer Vision, Graphics, and Image Processing*, 21(1):3–20, 1983.

[13] Guillermo D Canas, Yuriy Vasilyev, Yair Adato, Todd Zickler, Steven Gortler, and Ohad Ben-Shahar. A linear formulation of shape from specular flow. In *International Conference on Computer Vision (ICCV)*, 2009.

[14] Ayan Chakrabarti and Todd Zickler. Depth and deblurring from a spectrally-varying depth-of-field. In *European Conference on Computer Vision (ECCV)*, 2012.

[15] Manmohan Chandraker. What camera motion reveals about shape with unknown brdf. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

[16] Manmohan Chandraker. On shape and material recovery from motion. In *European Conference on Computer Vision (ECCV)*, 2014.

[17] Manmohan Chandraker. The information available to a moving observer on shape with unknown, isotropic brdfs. *Pattern Analysis and Machine Intelligence*, 38(7):1283–1297, 2016.

[18] Manmohan Chandraker, Jiamin Bai, and Ravi Ramamoorthi. A theory of photometric reconstruction for unknown isotropic reflectances. *Technical Report EECS-2010-176*, 2010.

[19] Manmohan Chandraker, Jiamin Bai, and Ravi Ramamoorthi. A theory of differential photometric stereo for unknown isotropic brdfs. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[20] Manmohan Chandraker, Jiamin Bai, and Ravi Ramamoorthi. On differential photometric reconstruction for unknown, isotropic brdfs. *Pattern Analysis and Machine Intelligence*, 35(12):2941–2955, 2013.

[21] Manmohan Chandraker, Dikpal Reddy, Yizhou Wang, and Ravi Ramamoorthi. What object motion reveals about shape with unknown brdf and lighting. In *Computer Vision and Pattern Recognition (CVPR)*, 2013.

[22] Jen-Hao Chang, B. V. K. Vijaya Kumar, and Aswin C. Sankaranarayanan. Towards multifocal displays with dense focal stacks. *SIGGRAPH Asia / ACM Trans. Graphics*, 37(6):1–13, 2018.

[23] N Ng Kuang Chern, Poo Aun Neow, and Marcelo H Ang. Practical issues in pixel-based autofocusing for machine vision. In *International Conference on Robotics and Automation (ICRA)*, 2001.

[24] Alessandro Chiuso, Roger Brockett, and Stefano Soatto. Optimal structure from motion: Local ambiguities and global estimates. *International Journal of Computer Vision*, 39(3):195–228, 2000.

[25] Steven A Cholewiak, Gordon D Love, Pratul P Srinivasan, Ren Ng, and Martin S Banks. Chromablur: Rendering chromatic eye aberration improves accommodation and realism. *ACM Transactions on Graphics (TOG)*, 36(6):210, 2017.

[26] James J Clark. Active photometric stereo. In *Computer Vision and Pattern Recognition (CVPR)*, 1992.

[27] T Collett. Stereopsis in toads. *Nature*, 267(5609):349, 1977.

[28] SP Collin, RV Hoskins, and JC Partridge. Tubular eyes of deep-sea fishes: A comparative study of retinal topography (part 1 of 2). *Brain, Behavior and Evolution*, 50(6):335–346, 1997.

[29] Pierre-Emile J Duhamel, Castor O Perez-Arancibia, Geoffrey L Barrows, and Robert J Wood. Biologically inspired optical-flow sensing for altitude control of flapping-wing microrobots. *IEEE/ASME Transactions on Mechatronics*, 18(2):556–568, 2013.

[30] John Ens and Peter Lawrence. A matrix based method for determining depth from focus. In *Computer Vision and Pattern Recognition (CVPR)*, 1991.

[31] Ahmet M Eskicioglu and Paul S Fisher. Image quality measures and their performance. *IEEE Transactions on Communications*, 43(12):2959–2965, 1995.

[32] Hany Farid and Eero P Simoncelli. Range estimation by optical differentiation. *Journal of the Optical Society of America A*, 15(7): 1777–1786, 1998.

[33] Lawrence Firestone, Kitty Cook, Kevin Culp, Neil Talsania, and Kendall Preston Jr. Comparison of autofocus methods for automated microscopy. *Cytometry: The Journal of the International Society for Analytical Cytology*, 12(3):195–206, 1991.

[34] Stephen D Fisher. *Complex variables*. Courier Corporation, 1999.

[35] Hassan Foroosh, Josiane B Zerubia, and Marc Berthod. Extension of phase correlation to subpixel registration. *IEEE Transactions on Image Processing*, 11(3):188–200, 2002.

[36] Jan-Mark Geusebroek, Frans Cornelissen, Arnold WM Smeulders, and Hugo Geerts. Robust autofocusing in microscopy. *Cytometry: The Journal of the International Society for Analytical Cytology*, 39(1):1–9, 2000.

[37] Joshua Gluckman and Shree K Nayar. Ego-motion and omnidirectional cameras. In *International Conference on Computer Vision (ICCV)*, 1998.

[38] Paul Grossmann. Depth from focus. *Pattern Recognition Letters*, 5(1): 63–69, 1987.

[39] Qi Guo, Emma Alexander, and Todd E Zickler. Focal track: Depth and accommodation with oscillating lens deformation. In *International Conference on Computer Vision (ICCV)*, 2017.

[40] Lindesay Harkness. Chameleons use accommodation cues to judge distance. *Nature*, 267(5609):346, 1977.

[41] Duane P Harland and Robert R Jackson. Portia perceptions: the umwelt of an araneophagic jumping spider. *Complex worlds from Simpler Nervous Systems*, pages 5–40, 2004.

[42] James Edward Heath, R Glenn Northcutt, and Robert P Barber. Rotational optokinesis in reptiles and its bearing on pupillary shape. *Zeitschrift für vergleichende Physiologie*, 62(1):75–85, 1969.

[43] David J Heeger and Allan D Jepson. Subspace methods for recovering rigid motion i: Algorithm and implementation. *International Journal of Computer Vision*, 7(2):95–117, 1992.

[44] Franz Stephan Helmli and Stefan Scherer. Adaptive shape from focus with an error estimation in light microscopy. In *International Symposium on Image and Signal Processing and Analysis (ISPA)*, 2001.

[45] Berthold K Horn and Brian G Schunck. Determining optical flow. In *1981 Technical Symposium East*. International Society for Optics and Photonics, 1981.

[46] Berthold KP Horn, Yajun Fang, and Ichiro Masaki. Time to contact relative to a planar surface. In *Intelligent Vehicles Symposium (IV)*, 2007.

[47] Berthold KP Horn, Yajun Fang, and Ichiro Masaki. Hierarchical framework for direct gradient-based time-to-contact estimation. In *Intelligent Vehicles Symposium (IV)*, 2009.

[48] Fred Hoyle. *The Black Cloud*. Penguin, 1957.

[49] Wei Huang and Zhongliang Jing. Evaluation of focus measures in multi-focus image fusion. *Pattern recognition letters*, 28(4):493–500, 2007.

[50] T-L Hwang, James J Clark, and Alan L Yuille. A depth recovery algorithm using defocus information. In *Computer Vision and Pattern Recognition (CVPR)*, 1989.

[51] Kenichi Kanatani. 3-d interpretation of optical flow by renormalization. *International Journal of Computer Vision*, 11(3):267–282, 1993.

[52] Sanjeev J Koppal, Ioannis Gkioulekas, Todd Zickler, and Geoffrey L Barrows. Wide-angle micro sensors for vision on a tight budget. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[53] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. In *Advances in Neural Information Processing Systems*, 2009.

[54] RHH Kröger, MCW Campbell, RD Fernald, and H-J Wagner. Multifocal lenses compensate for chromatic defocus in vertebrate eyes. *Journal of Comparative Physiology A*, 184(4):361–369, 1999.

[55] Eric Krotkov and J-P Martin. Range from focus. In *International Conference on Robotics and Automation (ICRA)*, 1986.

[56] MF Land. Image formation by a concave reflector in the eye of the scallop, pecten maximus. *The Journal of Physiology*, 179(1):138–153, 1965.

[57] MF Land. Movements of the retinae of jumping spiders (salticidae: Dendryphantinae) in response to visual stimuli. *Journal of Experimental Biology*, 51(2):471–493, 1969.

[58] MF Land. Structure of the retinae of the principal eyes of jumping spiders (salticidae: Dendryphantinae) in relation to visual optics. *Journal of Experimental Biology*, 51(2):443–470, 1969.

[59] Markus Lappe and Josef P Rauschecker. A neural network for the processing of optic flow from ego-motion in man and higher mammals. *Neural Computation*, 5(3):374–391, 1993.

[60] David N Lee. A theory of visual control of braking based on information about time-to-collision. *Perception*, (5):437–59, 1976.

[61] Eunsung Lee, Eunjung Chae, Hejin Cheong, Semi Jeon, and Joonki Paik. Depth-based defocus map estimation using off-axis apertures. *Optics Express*, 23(17):21958–21971, 2015.

[62] Je-Ho Lee, Kun-Sop Kim, Byung-Deok Nam, Jae-Chon Lee, Yong-Moo Kwon, and Hyoung-Gon Kim. Implementation of a passive automatic focusing algorithm for digital still camera. *Transactions on Consumer Electronics*, 41(3):449–454, 1995.

[63] Sang-Yong Lee, Yogendera Kumar, Ji-Man Cho, Sang-Won Lee, and Soo-Won Kim. Enhanced autofocus algorithm using robust focus measure and fuzzy reasoning. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(9):1237–1246, 2008.

[64] Sang-Yong Lee, Jae-Tack Yoo, Yogendera Kumar, and Soo-Won Kim. Reduced energy-ratio measure for robust autofocusing in digital camera. *IEEE Signal Processing Letters*, 16(2):133–136, 2009.

[65] Anat Levin. Analyzing depth from coded aperture sets. In *European Conference on Computer Vision (ECCV)*, 2010.

[66] Anat Levin, Rob Fergus, Frédo Durand, and William T Freeman. Image and depth from a conventional camera with a coded aperture. In *ACM Transactions on Graphics (TOG)*, 2007.

[67] H Christopher Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133, 1981.

[68] Hugh Christopher Longuet-Higgins, Kvetoslav Prazdny, et al. The interpretation of a moving retinal image. *Proceedings of the Royal Society London B B*, 208(1173):385–397, 1980.

[69] Bruce D Luccas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. *International Joint Conferences on Artificial Intelligence (IJCAI)*, 1981.

[70] Aamir Saeed Malik and Tae-Sun Choi. A novel algorithm for estimation of depth map using image focus for 3d shape recovery in the presence of noise. *Pattern Recognition*, 41(7):2200–2225, 2008.

[71] Karunyakanth Mandapaka, Randy C Morgan, and Elke K Buschbeck. Twenty-eight retinas but only twelve eyes: An anatomical analysis of the larval visual system of the diving beetle thermonectus marmoratus (coleoptera: Dytiscidae). *Journal of Comparative Neurology*, 497(2): 166–181, 2006.

[72] Amrita Mazumdar, Armin Alaghi, Jonathan T Barron, David Gallup, Luis Ceze, Mark Oskin, and Steven M Seitz. A hardware-friendly bilateral solver for real-time virtual reality video. In *Proceedings of High Performance Graphics*. ACM, 2017.

[73] Gil Menda, Paul S Shamble, Eyal I Nitzany, James R Golden, and Ronald R Hoy. Visual perception in the brain of a jumping spider. *Current Biology*, 24(21):2580–2585, 2014.

[74] Daniel Miau, Oliver Cossairt, and Shree K Nayar. Focal sweep videography with deformable optics. In *International Conference on Computational Photography (ICCP)*, 2013.

[75] Rashid Minhas, Abdul A Mohammed, QM Jonathan Wu, and Maher A Sid-Ahmed. 3d shape from focus and depth map computation using steerable filters. In *International Conference Image Analysis and Recognition*, 2009.

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

[77] Christopher J Murphy and Howard C Howland. On the gekko pupil and scheiner's disc. *Vision Research*, 26(5):815–817, 1986.

[78] Takashi Nagata, Mitsumasa Koyanagi, Hisao Tsukamoto, Shinjiro Saeki, Kunio Isono, Yoshinori Shichida, Fumio Tokunaga, Michiyo Kinoshita, Kentaro Arikawa, and Akihisa Terakita. Depth perception from image defocus in a jumping spider. *Science*, 335(6067):469–471, 2012.

[79] Hari N Nair and Charles V Stewart. Robust focus ranging. In *Computer Vision and Pattern Recognition (CVPR)*, 1992.

[80] Shree K Nayar and Yasuo Nakagawa. Shape from focus: An effective approach for rough surfaces. In *International Conference on Robotics and Automation (ICRA)*, 1990.

[81] Ren Ng. Fourier slice photography. In *ACM Transactions on Graphics (TOG)*, 2005.

[82] Michael Oren and Shree K Nayar. A theory of specular surface geometry. *International Journal of Computer Vision*, 24(2):105–124, 1997.

[83] Karl Pauwels and Marc M Van Hulle. Optimal instantaneous rigid motion estimation insensitive to local minima. *Computer Vision and Image Understanding*, 104(1):77–86, 2006.

[84] José Luis Pech-Pacheco, Gabriel Cristóbal, Jesús Chamorro-Martinez, and Joaquín Fernández-Valdivia. Diatom autofocusing in brightfield microscopy: a comparative study. In *International Conference on Pattern Recognition*, 2000.

[85] Alex Paul Pentland. A new sense for depth of field. *Pattern Analysis and Machine Intelligence*, (4):523–531, 1987.

[86] John A Perrone. Model for the computation of self-motion in biological systems. *JOSA A*, 9(2):177–194, 1992.

[87] Said Pertuz, Domenec Puig, and Miguel Angel Garcia. Analysis of focus measure operators for shape-from-focus. *Pattern Recognition*, 46(5): 1415–1432, 2013.

[88] AN Rajagopalan and Subhasis Chaudhuri. Optimal selection of camera parameters for recovery of depth from defocused images. In *Computer Vision and Pattern Recognition (CVPR)*, 1997.

[89] Puig D. Rashwan, H. A. and M. A. Garcia. On improving the robustness of differential optical flow. In *International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011.

[90] Florian Raudies and Heiko Neumann. A review and evaluation of methods estimating ego-motion. *Computer Vision and Image Understanding*, 116(5):606–633, 2012.

[91] JH Rieger and DT Lawton. Processing differential image motion. *JOSA A*, 2(2):354–359, 1985.

[92] Lina SV Roth, Linda Lundström, Almut Kelber, Ronald HH Kröger, and Peter Unsbo. The pupils and optical systems of gecko eyes. *Journal of Vision*, 9(3):27–27, 2009.

[93] Walter Rudin. *Functional analysis*. McGraw-Hill, 1991.

[94] Megan J Russell and Tania S Douglas. Evaluation of autofocus algorithms for tuberculosis microscopy. In *International Conference on Engineering in Medicine and Biology Society*, 2007.

[95] Andrés Santos, C Ortiz de Solórzano, Juan José Vaquero, JM Pena, Norberto Malpica, and F Del Pozo. Evaluation of autofocus functions in molecular cytogenetic analysis. *Journal of Microscopy*, 188(3):264–272, 1997.

[96] Yoav Y Schechner and Nahum Kiryati. Depth from defocus vs. stereo: How different really are they? *International Journal of Computer Vision*, 39 (2):141–162, 2000.

[97] Omid Shakernia, René Vidal, and Shankar Sastry. Omnidirectional egomotion estimation from back-projection flow. In *IEEE Workshop Omnidirectional Vision*, 2003.

[98] Chun-Hung Shen and Homer H Chen. Robust focus measure for low-contrast images. In *International Conference on Consumer Electronics (ICCE)*, 2006.

[99] Claudiu A Stan. Liquid optics: Oscillating lenses focus fast. *Nature Photonics*, 2(10):595, 2008.

[100] Annette Stowasser, Alexandra Rapaport, John E Layne, Randy C Morgan, and Elke K Buschbeck. Biological bifocal lenses with image separation. *Current Biology*, 20(16):1482–1486, 2010.

[101] Alexander L Stubbs and Christopher W Stubbs. Spectral discrimination in color blind animals via chromatic aberration and pupil shape. *Proceedings of the National Academy of Sciences*, 113(29):8206–8211, 2016.

[102] Murali Subbarao. Parallel depth recovery by changing camera parameters. In *International Conference on Computer Vision (ICCV)*, 1988.

[103] Murali Subbarao and Gopal Surya. Depth from defocus: A spatial domain approach. *International Journal of Computer Vision*, 13(3):271–294, 1994.

[104] Murali Subbarao, Tae-Sun Choi, and Arman Nikzad. Focusing techniques. *Optical Engineering*, 32(11):2824–2837, 1993.

[105] Muralidhara Subbarao. Efficient depth recovery through inverse optics. In *Machine Vision for Inspection and Measurement*, pages 101–126. Elsevier, 1989.

[106] Yu Sun, Stefan Duthaler, and Bradley J Nelson. Autofocusing in computer microscopy: selecting the optimal focus algorithm. *Microscopy Research and Technique*, 65(3):139–149, 2004.

[107] Supasorn Suwajanakorn, Carlos Hernandez, and Steven M Seitz. Depth from focus with your mobile phone. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[108] Huixuan Tang, Scott Cohen, Brian Price, Stephen Schiller, and Kiriakos N Kutulakos. Depth from defocus in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[109] Andrea Thelen, Susanne Frey, Sven Hirsch, and Peter Hering. Improvements in shape-from-focus for holographic reconstructions with regard to focus operators, neighborhood-size, and height value interpolation. *IEEE Transactions on Image Processing*, 18(1):151–157, 2009.

[110] Ashok Veeraraghavan, Ramesh Raskar, Amit Agrawal, Ankit Mohan, and Jack Tumblin. Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing. In *ACM Transactions on Graphics (TOG)*, 2007.

[111] Neal Wadhwa, Rahul Garg, David E Jacobs, Bryan E Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T Barron, Yael Pritch, and Marc Levoy. Synthetic depth-of-field with a single-camera mobile phone. *ACM Transactions on Graphics (TOG)*, 2018.

[112] Hermann Wagner and Frank Schaeffel. Barn owls (tyto alba) use accommodation as a distance cue. *Journal of Comparative Physiology A*, 169(5):515–521, 1991.

[113] Ting-Chun Wang, Manmohan Chandraker, Alexei A Efros, and Ravi Ramamoorthi. Svbrdf-invariant shape and reflectance estimation from light-field cameras. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[114] Masahiro Watanabe and Shree K Nayar. Rational filters for passive depth from defocus. *International Journal of Computer Vision*, 27(3):203–225, 1998.

[115] Chong-Yaw Wee and Raveendran Paramesran. Measure of image sharpness using eigenvalues. *Information Sciences*, 177(12):2533–2552, 2007.

[116] Chong-Yaw Wee and Raveendran Paramesran. Comparative analysis of eigenvalues-based and tchebichef moments-based focus measures. In *International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, 2008.

[117] Chong-Yaw Wee and Raveendran Paramesran. Image sharpness measure using eigenvalues. In *International Conference on Signal Processing*, 2008.

[118] Walter Thomson Welford. Use of annular apertures to increase focal depth. *JOSA*, 50(8):749–753, 1960.

[119] Andrew P Witkin. Scale-space filtering. *International Joint Conferences on Artificial Intelligence (IJCAI)*, 1983.

[120] Tao Xian and Murali Subbarao. Depth-from-defocus: Blur equalization technique. In *International Society for Optics and Photonics (SPIE)*, 2006.

[121] Hui Xie, Weibin Rong, and Lining Sun. Wavelet-based focus measure and 3-d surface reconstruction method for microscopy images. In *Intelligent Robots and Systems (IROS)*, 2006.

[122] Ge Yang and Bradley J Nelson. Wavelet-based autofocusing and unsupervised segmentation of microscopic images. In *Intelligent Robots and Systems (IROS)*, 2003.

[123] Pew Thian Yap and P Raveendran. Image focus measure based on chebyshev moments. *IEE Proceedings-Vision, Image and Signal Processing*, 151(2):128–136, 2004.

[124] Fumihito Yasuma, Tomoo Mitsunaga, Daisuke Iso, and Shree K Nayar. Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum. *IEEE Transactions on Image Processing*, 19 (9):2241–2253, 2010.

[125] Marina Zannoli, Gordon D Love, Rahul Narain, and Martin S Banks. Blur and the perception of depth at occlusions. *Journal of Vision*, 16(6):17–17, 2016.

[126] Ruo Zhang, Ping-Sing Tsai, James Edwin Cryer, and Mubarak Shah. Shape-from-shading: a survey. *Pattern Analysis and Machine Intelligence*, 21(8):690–706, 1999.

[127] Tong Zhang and Carlo Tomasi. Fast, robust, and consistent camera motion estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 1999.

[128] Jiang Yu Zheng and Akio Murata. Acquiring a complete 3d model from specular motion under the illumination of circular-shaped light sources. *Pattern Analysis and Machine Intelligence*, (8):913–920, 2000.

[129] Jiang Yu Zheng, Yoshihiro Fukagawa, and Norihiro Abe. Shape and model from specular motion. In *International Conference on Computer Vision (ICCV)*, 1995.

[130] Changyin Zhou, Stephen Lin, and Shree Nayar. Coded aperture pairs for depth from defocus. In *International Conference on Computer Vision (ICCV)*, 2009.

[131] Changyin Zhou, Stephen Lin, and Shree K Nayar. Coded aperture pairs for depth from defocus and defocus deblurring. *International Journal of Computer Vision*, 93(1):53–72, 2011.

[132] Xinhua Zhuang, Robert M Haralick, and Yunxin Zhao. From depth and optical flow to rigid body motion. In *Computer Vision and Pattern Recognition (CVPR)*, 1988.

A
Alternative Derivations of the Constraint

Setting aside the question of uniqueness, this appendix assumes Gaussian blur and provides two additional derivations of the defocus brightness constancy constraint. These alternatives to section 3.2.1 may provide additional intuition for how the constraint arises. One of these derivations is based on a truncated Taylor expansion, mirroring a common derivation for linearized optical flow. The other is based on sinusoidal textures, as illustrated in figure 3.2.1 and analyzed in section 3.2.2 for inherent sensitivity.

A.1 From Taylor Expansion

Following the well-known Taylor series derivation for differential optical flow, consider the difference in intensity at a pixel between a pair of images taken a time step Δt apart. Recall that the brightness of the underlying sharp texture does not change, but the change in defocus blur must be accounted for to use the real-aperture images.

To do so, assume Gaussian blur kernels k,

$$k(x, y, \sigma) = \frac{e^{-\frac{ax^2+bxy+cy^2}{\sigma^2}}}{\sigma^2}, \tag{A.1}$$

and define a reblurring filter b that takes narrow Gaussians to wider Gaussians under spatial convolution:

$$k(x, y, \sigma_2) = b\left(x, y, \sigma_1, \sigma_2\right) * k(x, y, \sigma_1)\,. \tag{A.2}$$

Recall that for Gaussian blur this reblurring filter takes the form of another Gaussian,

$$b(x, y, \sigma_1, \sigma_2) = k(x, y, \sqrt{\sigma_2^2 - \sigma_1^2}), \tag{A.3}$$

which under no change in blur collapses to a Dirac delta,

$$b(x, y, \sigma_1, \sigma_1) = k(x, y, 0) = \delta(x, y). \tag{A.4}$$

The all-in-focus unchanging-texture brightness constancy constraint states that for the pinhole image P,

$$P(x + \Delta x, y + \Delta y, t + \Delta t) = P(x, y, t), \tag{A.5}$$

with features moving from (x, y) to $(x + \Delta x, y + \Delta y)$ on the image. This remains true after convolving both sides of the constraint by the same Gaussian k, with blur scale $\sigma(t + \Delta t)$:

$$k(x, y, \sigma(t + \Delta t)) * P(x + \Delta x, y + \Delta y, t + \Delta t) = k(x, y, \sigma(t + \Delta t)) * P(x, y, t). \tag{A.6}$$

Without loss of generality, set the sign of Δt so that that $\sigma(t + \Delta t) > \sigma(t)$, and note that the above constraint relates the image taken at time $t + \Delta t$ to a reblurred version of the image taken at time t:

$$\begin{aligned} I(x + \Delta x, y + \Delta y, t + \Delta t) = b\left(x, y, \sigma(t), \left(\frac{Z(t + \Delta t)}{Z(t)} - \frac{f(t + \Delta t)}{f(t)}\right)\sigma(t + \Delta t)\right) \\ * I(x, y, t), \end{aligned} \tag{A.7}$$

where the $\frac{Z(t+\Delta t)}{Z(t)} - \frac{f(t+\Delta t)}{f(t)}$ term accounts for the change in magnification between images.

Taking the Taylor expansion of both sides and dropping terms above first order produces the approximation

$$\begin{aligned}I(x,y,t)+I_x\Delta x+I_y\Delta y+I_t\Delta t \;\approx\;& \delta(x,y)*I(x,y,t)+ \\ & \Delta t\left(\sigma_t+\left(\frac{Z_t}{Z}-\frac{f_t}{f}\right)\sigma\right)b_{\sigma_2}(x,y,\sigma,\sigma)*I(x,y,t).\end{aligned} \tag{A.8}$$

Subtracting the $I(x,y,t)$ term from each side, dividing by Δt, and noting that the reblurring filter collapses to a Dirac delta as

$$b_{\sigma_2}(x,y,\sigma_1,\sigma_2) = \sigma_2\Sigma^2\nabla^2_{\boldsymbol{\Sigma}}b(x,y,\sigma_1,\sigma_2) \tag{A.9}$$

$$\rightarrow b_{\sigma_2}(x,y,\sigma,\sigma) = \sigma\Sigma^2\nabla^2_{\boldsymbol{\Sigma}}\delta(x,y), \tag{A.10}$$

the approximate constraint becomes

$$I_x\frac{\Delta x}{\Delta t}+I_y\frac{\Delta y}{\Delta t}+I_t\approx\left(\sigma_t+\left(\frac{Z_t}{Z}-\frac{f_t}{f}\right)\sigma\right)\sigma\Sigma^2\nabla^2_{\boldsymbol{\Sigma}}I. \tag{A.11}$$

In the limit as Δt approaches zero, the ratios $\left(\frac{\Delta x}{\Delta t},\frac{\Delta y}{\Delta t}\right)$ approach the optical flow vector $(\dot{x},\dot{y})$. Separating this flow into translation and magnification terms as in section 3.2 produces the defocus brightness constancy constraint.

A.2 From Single-Frequency Texture

For general sinusoidal texture T,

$$T(u,v) = B_0 \sin(\omega_u u + \omega_v v + \varphi_0), \tag{A.12}$$

a pinhole camera will record the image P,

$$\begin{aligned} P(x,y,t) &= B_0 \sin(\omega_x(t)x + \omega_y(t)y + \varphi(t)), && \text{(A.13)} \\ \omega_x &= -\frac{Z(t)}{f(t)}\omega_u, && \text{(A.14)} \\ \omega_y &= -\frac{Z(t)}{f(t)}\omega_v, && \text{(A.15)} \\ \varphi &= -\omega_u X(t) - \omega_v Y(t) + \varphi_0. && \text{(A.16)} \end{aligned}$$

Under Gaussian blur as in equation (A.1), frequency and phase will not change but amplitude will:

$$I(x,y,t) = B(t)\sin(\omega_x x + \omega_y y + \varphi), \tag{A.17}$$

$$B(t) = \max_{\varphi}(k * P) = B_0 \mathcal{F}[k](\omega_x, \omega_y) = \frac{2\pi B_0}{\sqrt{4ac - b^2}} e^{-\frac{\Sigma^2\sigma^2}{2}(c\omega_x^2 - b\omega_x\omega_y + a\omega_x^2)}. \tag{A.18}$$

The derivatives of this image are as follows, where dots indicate time derivatives:

$$\begin{aligned} I_x &= \omega_x B \cos(\omega_x x + \omega_y y + \varphi), && \text{(A.19)} \\ I_y &= \omega_y B \cos(\omega_x x + \omega_y y + \varphi), && \text{(A.20)} \\ I_{xx} &= -\omega_x^2 B \sin(\omega_x x + \omega_y y + \varphi), && \text{(A.21)} \\ I_{xy} &= -\omega_x \omega_y B \sin(\omega_x x + \omega_y y + \varphi), && \text{(A.22)} \\ I_{yy} &= -\omega_y^2 B \sin(\omega_x x + \omega_y y + \varphi), && \text{(A.23)} \\ I_t &= (\dot{\varphi} + \dot{\omega}_x x + \dot{\omega}_y y) B \cos(\omega_x x + \omega_y y + \varphi) \\ &\quad + \dot{B} \sin(\omega_x x + \omega_y y + \varphi), && \text{(A.24)} \end{aligned}$$

so that

$$\begin{aligned} I_t =& - \frac{\omega_u \dot{X}}{\omega_x} I_x - \frac{\omega_v \dot{Y}}{\omega_y} I_y + \frac{\dot{\omega}_x x}{\omega_x} I_x + \frac{\dot{\omega}_y y}{\omega_y} I_y \\ & + \frac{\dot{B}}{B} \frac{c I_{xx} - b I_{xy} + a I_{yy}}{\left(-c\omega_x^2 + b\omega_x\omega_y - a\omega_y^2\right)} \end{aligned} \tag{A.25}$$

$$\begin{aligned} =& \frac{f}{Z} X_t I_x + \frac{f}{Z} Y_t I_y + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) x I_x + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) y I_y \\ & + \left(\sigma_t + \left(\frac{Z_t}{Z} - \frac{f_t}{f} \right) \sigma \right) \sigma \Sigma^2 \nabla_{\mathbf{\Sigma}}^2 I. \end{aligned} \tag{A.26}$$

By the linearity of convolution and differentiation, this equation holds for all sum-of-sinusoid textures, so that the defocus brightness constancy constraint applies to any natural texture.

B
Depth from Differential Gaussian Aperture Modification

Chapter 3 describes in detail the effect of changes in blur scale, but it does not discuss the effect of changing the aperture filter between shots, another common way of of generating a defocus change for depth inference. This appendix will describe how a differential change in a Gaussian aperture filter can be used to extract scene information. There are two key differences between the results in this appendix and those in chapter 3. The first is that the proof of Gaussian uniqueness does not apply to this scenario, which permits a differential shift approach that works for any filter. The second is that, unlike the unambiguous results from the defocus brightness constancy constraint, there is a side of focal plane ambiguity that arises when recovering depth from such situations as an overall aperture scaling, a stretching along one axis, rotation of an anisotropic Gaussian, or a combination of all of these.

In this appendix, the Gaussian filter will be generalized beyond its covariance (a, b, c) to include a variable mean (μ^x, μ^y) and overall transmittance factor d. That is, it can now stretch, rotate, shift, and dim, according to:

$$\kappa(x, y, t) = e^{-a(t)(x-\mu^x(t))^2 - b(t)(x^{\mu}_{x}(t)(y-\mu^y(t)) - c(y-\mu^y(t))^2 - d(t)}. \tag{B.1}$$

From this filter, a corresponding post-processing operation for depth recovery can be derived. Recall that, with no scene motion or change in camera parameters, image change I_t is simply $k_t * P$. The goal is to recover a depth-dependent scalar v and a depth-blind post-processing operator $m(x, y)$ such that

$$I_t = v\, m * I \tag{B.2}$$

for any underlying pinhole image P. This universality requirement implies that the operator m must take the blur kernel k to something proportional to its time derivative:

$$\begin{aligned} k_t * P =& v\, m * k * P, \;\; \forall P \qquad \text{(B.3)} \\ \rightarrow k_t =& v\, m * k. \qquad \text{(B.4)} \end{aligned}$$

In the frequency domain, this convolution takes the form of a multiplication, so that, with $\mathcal{F}$ indicating the Fourier transform,

$$v\, m = \mathcal{F}^{-1}\left[\mathcal{F}[k_t]/\mathcal{F}[k]\right]. \tag{B.5}$$

Combining equations B.1 and B.5, the general form of vm can be seen to contain the blur scale σ raised to the zeroth, first, and second powers:

$$v\, m = \left(\frac{\Sigma^2}{2}(bb_t - 2ac_t - 2ca_t) - d_t\right)\delta - \sigma(\mu_t^x \partial_x + \mu_t^y \partial_y) + \sigma^2 \nabla^2_{\mathbf{\Sigma},t}, \tag{B.6}$$

where the notation $\nabla^2_{\mathbf{\Sigma},t}$ indicates a new warped Laplacian, which now depends not only on the Gaussian widths a, b, c but also on their time derivatives:

$$\begin{aligned} \nabla^2_{\mathbf{\Sigma},t} = & -\frac{\Sigma^4}{4}\left(4c^2 a_t - 2bcb_t + b^2 c_t\right)\partial_{xx} \\ & -\frac{\Sigma^4}{4}\left(-4bca_t + b^2 b_t + 4acb_t - 4abc_t\right)\sigma^2\partial_{xy} \\ & -\frac{\Sigma^4}{4}\left(b^2 a_t - 2abb_t + 4a^2 c_t\right)\sigma^2\partial_{yy}. \end{aligned} \tag{B.7}$$

The multiple powers of σ in equation B.6 are a problem because it makes the expression for vm impossible to split into the product of a term v that depends on σ with a function $m(x, y)$ that does not. However, this equation can still reveal depth with some manipulation and by respecting certain physical limitations.

The first step towards a useful constraint is removing the first, depth-independent Dirac delta term from vm. This term reflects the overall change in light efficiency of the aperture as it simultaneously shrinks or broadens (through a_t, b_t, and c_t) while brightening or dimming (through d_t). Because the lightness change ΔL, with

$$\Delta L = \frac{\Sigma^2}{2}(bb_t - 2ac_t - 2ca_t) - d_t, \tag{B.8}$$

depends only on known quantities, it can be accounted for with a simple pre-processing step:

$$I_t - (\Delta L)\, I = v'\, m' * I, \tag{B.9}$$

where $v'm'$ is a new operator that now contains only σ and σ^2 terms:

$$v'm' = -\,\sigma(\mu_t^x \partial_x + \mu_t^y \partial_y) - \sigma^2 \nabla^2_{\Sigma,t}. \tag{B.10}$$

The combination of σ powers in equation B.10 still prevents the desired depth-blind factoring into a σ-dependent v' and a σ-independent m'. It points to an underlying limitation on the kinds of aperture changes that can reveal depth in the desired way. Specifically, it leaves two possibilities: 1) when the σ^2 term vanishes, then σ and therefore depth can be recovered from the first spatial image derivatives, and 2) when the σ term vanishes, then σ^2 and therefore depth up to a side of focal plane ambiguity can be recovered from the second spatial image derivatives.

In the first case, $a_t = b_t = c_t = 0$, and the aperture filter is a Gaussian that undergoes a lateral shift (μ_t^x, μ_t^y). Then the light-efficiency-corrected image change $I_t - (\Delta L)I$ takes the form

$$I_t + d_t I = -\,\sigma\left(\mu_t^x I_x + \mu_t^y I_y\right), \tag{B.11}$$

which reveals depth directly through the equation

$$Z = \left(\frac{1}{Z_f} - \frac{I_t + d_t I}{f\mu_t^x I_x + f\mu_t^y I_y}\right)^{-1}. \tag{B.12}$$

This is essentially a differential version of depth from stereo, where the baseline is the σ-magnified filter shift. Depth from defocus and stereo are known to share fundamental similarities [96], and this approach has been used for pinhole [51] and pillbox apertures [111], and would work in the same way for any filter κ.

In the second case, depth can be recovered up to a side of focal plane ambiguity as long as the Gaussian does not shift laterally ($\mu_t^x = \mu_t^y = 0$). Then, image values are constrained by

$$I_t - (\Delta L)I = -\sigma^2 \nabla^2_{\mathbf{\Sigma},t} I, \tag{B.13}$$

and per-pixel depth is recovered, up to a side of focal plane ambiguity, by

$$Z = \left(\frac{1}{Z_f} \pm \sqrt{\frac{I_t - (\Delta L)I}{-f^2 \nabla^2_{\mathbf{\Sigma},t} I}} \right)^{-1}. \tag{B.14}$$

This generalizes the differential change in aperture size used in [32] and [103].

Colophon

THIS THESIS WAS TYPESET using LaTeX, originally developed by Leslie Lamport and based on Donald Knuth's TeX. The body text is set in 11 point Arno Pro, designed by Robert Slimbach in the style of book types from the Aldine Press in Venice, and issued by Adobe in 2007. A template, which can be used to format a PhD thesis with this look and feel, has been released under the permissive MIT (X11) license, and can be found online at github.com/suchow/ or from the author at suchow@post.harvard.edu.