Simulation of Vision through an Actual Human Optical System

by

Woojin Matthew Yu

B.S.E. (University of Pennsylvania) 1998

A thesis submitted in partial satisfaction of the
requirements for the degree of
Master of Science

in

Engineering - Electrical Engineering and Computer Sciences

in the

GRADUATE DIVISION
of the
UNIVERSITY of CALIFORNIA, BERKELEY

Committee in charge:

Dr. Brian A. Barsky, Chair
Dr. Stanley A. Klein
Dr. David A. Forsyth

Fall 2001

The thesis of Woojin Matthew Yu is approved:

Chair Date

Date

Date

University of California, Berkeley

Fall 2001

Simulation of Vision through an Actual Human Optical System

by

Woojin Matthew Yu

Abstract

Simulation of Vision through an Actual Human Optical System

by

Woojin Matthew Yu

Master of Science in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Dr. Brian A. Barsky, Chair

This thesis presents a software system that simulates vision. One application is for those who suffer from ocular disorders. The development of the system delineates interesting problems in two different fields: computer graphics and optometry. The problem of using the human visual system as a model to render a photorealistic image is addressed. To simulate a patient's vision, the aberrations of light rays through the specific visual system needs to be measured and this is accomplished by a Shack–Hartmann device. Using the data from this device, algorithm constructs a wavefront, which is ultimately used to extrapolate the detailed information of the visual system. Along with this wavefront information, the system takes as inputs a perfectly sharp image to be blurred, together with the depth information of the image. Given the focusing distance as a parameter, the system uses the depth value supplied for each pixel to produce an appropriately blurred image that reproduces the vision perceived through the visual system being investigated. The simulation will

aid physicians to learn more about disorders and will help educate people about both the effects and side effects of corneal refractive surgeries, which are becoming popular.

Dr. Brian A. Barsky
Thesis Committee Chair

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Photo-realism in simulations has been pursued in the computer science community ever since the beginning of graphics, but even after thirty years, the process for rendering does not model the optics of camera nor the correct anatomy of human eyes. With our new algorithm, we wish to produce a simulation that not only achieves photo-realism, but that also achieves physical realism. The study uses the advancement in computer graphics to simulate the actual visual system in an effort to aid the fields of optometry and ophthalmology.

To briefly summarize our research, we built a software system that adds precisely-calculated blurs to a perfectly sharp image. It may sound simple, but introducing human optics, determining how the distortion is introduced, and how these "blurs" are calculated are integral parts of the research that are more complicated than it might seem. The algorithm needs to introduce distortions to a sharp image because the visual perception of the real world consists of both representations that are clearly in focus as well as those

that are out of focus. The introduction of optics and the details of patients' visual systems to existing simulations are the main motivation for the development of our system. The algorithm uses a set of Point Spread Functions (from here on, this will be referred to as PSF). A PSF describes how a point light source is smeared, and a set of PSF's are used to build properly distorted images. A perfectly sharp 3-D scene is provided along with the depth information of every pixel in the image, which is generated by a 3-D ray tracer that rendered the original picture. The Fast Fourier Transform (FFT) algorithm is used to perform convolution on discrete layers of pixels to create distortions. After blurring separate layers of image, they are merged into a single image, yielding the final product.

Calculating a point spread function requires a lot of computation time and hardware resources primarily because it needs to simulate and perform calculations on hundreds of thousands of light rays. Our research does not provide a fast algorithm to compute a PSF, but it does provide a way to convolve much faster. The algorithm can calculate a set of PSF's and store them on a secondary storage device. A set of PSF's represents a particular visual system and the set can be used to blur many different images that are viewed with the same visual system. Exploiting the pre-computed PSF, convolution can occur quickly without wasting system resources for recomputation.

The ultimate goal of this research is to simulate the vision of individuals. Knowing the behavior of light rays in an individual's human optical system, it is possible to construct images that are visual representations observed through the visual system being investigated. This enables the generation of images that can demonstrate specific defects in what a patient sees instead of settling for a more commonly accepted notion that their

visual imperfection is limited to a simple blur. This will promote better understanding of ocular disorders. The most practical use of the system is demonstrated in the case of the increasingly popular laser surgeries that permanently corrects for visual defects. Contrary to popular belief, these corrective surgeries do not always result in the flawless vision that patients expect. As the popularity of corneal refractive surgeries grows, it would be of significant advantage to enable the potential patients to understand the risks of side effects, including severe halo effect, visual fluctuations, and glare, which significantly affect people while driving at night. The potential candidates of the operation could be given an opportunity to preview the post-operative vision and could decide whether it is something that they truly need or want.

Chapter 2

Background

2.1 Anatomy

Humans and other vertebrates perceive images produced by light rays that pass through the cornea and crystalline lens. The iris is a structure located between the cornea and lens that determines the eye color and the central opening of this structure is called pupil. The size of the pupil changes according to the amount of light energy available in the surroundings to allow different amount of light to enter the eye, much like how aperture controls light level inside a camera. The pupil size decreases to prevent too much light from entering the eye during the day, and increases dramatically at night because more light rays need to be allowed on the retina. Ultimately, the rays focus to form an image on the retina where photoreceptors are located [17, 15].

The photoreceptors are classified according to their sensitivity: cone receptors are used to detect colors and rod receptors are used to detect overall brightness in the field of vision. The human retina has about 4.5 million cones and 91 million rods [17]. The cone

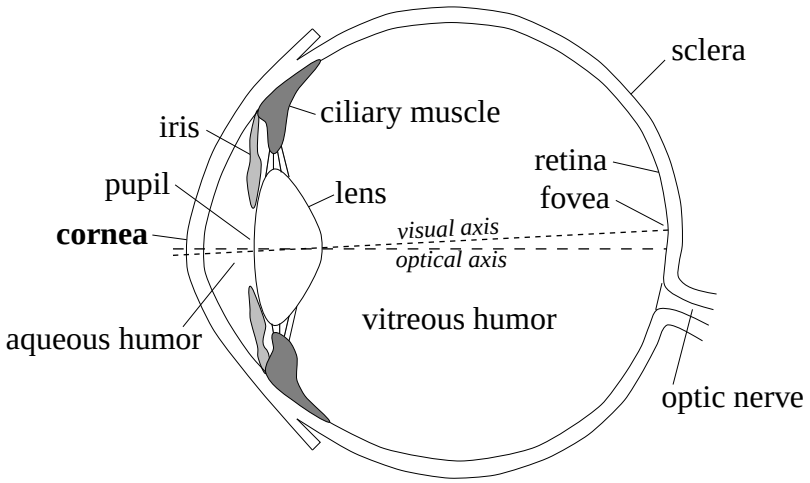


Figure 2.1: Human Eye. (Reprinted with permission from [8].)

cells are only responsive under bright conditions, which is why they are used for observing details. The rod cells do not distinguish color, but they respond under a minimal amount of light, making them very useful for night vision. The density of cone cells far exceeds the density of rod cells at the center of retina, known as the fovea, but the rod cells dominate in the periphery of the retina. Cones are divided into three groups according to their pigments. Each group is sensitive to a different range of frequencies of the electromagnetic spectrum and the mixed stimuli between them create a wide range of colors that vertebrates can distinguish. The nerves attached to the receptors then transmit these stimuli to the visual cortex of the brain where further processing occurs for correct interpretation.

The outermost surface of the eye is called cornea and it is the only part of the eye that can be transplanted. Its primary function is to bend light rays before they reach the retina. Although the cornea is clear and seems to lack substance, it is actually a highly organized group of cells and proteins. Lacking the regular blood vessels, it receives nour-

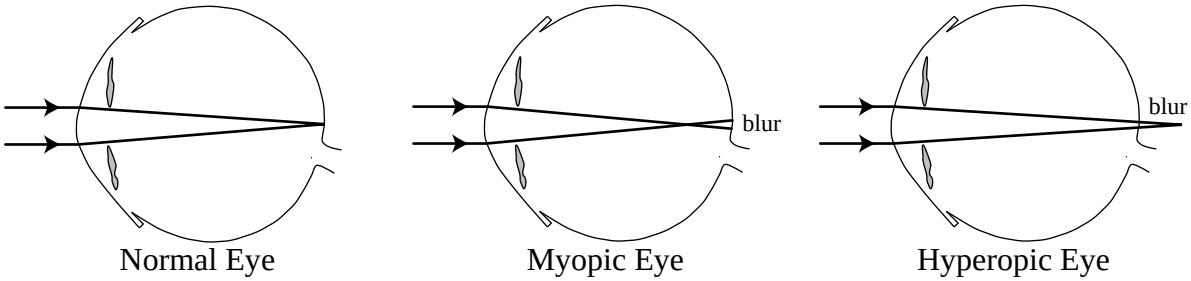


Figure 2.2: Difference between the normal, myopic and hyperopic conditions. (Reprinted with permission from [8].)

ishment from the tears and aqueous humor. Cornea is a five-layered transparent membrane that admits light into the interior of the eye and it is where two thirds of the total refraction occurs. For this reason, it is where majority of the ocular modifications takes place.

The notation of visual acuity is written as a fraction, with normal vision being 20/20. At a 20 foot distance, a person with normal vision should be able to read the small 20/20 line on an eye chart. The larger the second number is, the worse is the vision. A person with 20/200 vision would have to come up to 20 feet to see a letter that a person with normal vision could see at 200 feet. If the vision is 20/10, it means that the vision is better than normal. A person with 20/10 vision can read a letter at 20 feet that a person with normal vision would have to come up to 10 feet to read.

2.2 Imperfect Vision

About 157 million people in the U.S., which corresponds to about 58 percent of the population, wear eyeglasses or contact lenses [16]. Refractive errors occur when the shape of the cornea becomes irregular. When the cornea is of normal shape and curvature, it focuses light on the retina with precision, but with an irregularly shaped cornea, light

rays bend imperfectly, producing the blurry vision that many of us experience without the visual aids.

Myopia (Nearsightedness) The eye is too long for its focusing system. Myopic individuals have a natural advantage seeing up close. When the cornea is curved too much, faraway objects will appear blurry because they are focused in front of the retina.

Hyperopia (Farsightedness) is the exact opposite of myopia and distant objects appear clear and close-up objects appear blurry. Under this condition, the image forms behind the retina.

Astigmatism is a condition in which the uneven curvature of the cornea distorts both distant and near objects. With astigmatism, the cornea is curved in one direction more than the other, making it shaped like a football.

Ocular disorders like keratoconus, astigmatism, and macular degeneration can also be simulated using the algorithm. Keratoconus is characterized by progressive thinning and steepening of the central cornea. As the cornea steepens and thins, the patient experiences a deterioration in vision, which can be mild or severe depending on the amount of affected corneal tissue. It causes blurring and distortion of vision, and increased glare and sensitivity to light. As the condition progresses, the cornea bulges more and vision may become more distorted. In a small number of cases, the cornea will swell and cause a sudden and significant decrease in vision. Macular degeneration is the imprecise historical name given to a very poorly understood group of diseases that cause cells in the macular zone (near the fovea) of the retina to malfunction or lose function. The macula is a collection of cells

that line the retina that are extremely photosensitive. The result is debilitating loss of vital central of detail vision [10].

2.3 Refractive Surgeries

Radial Keratotomy(RK) is a surgical procedure that reduces myopia. Radial or spoke-like incisions are made into the cornea to structurally weaken and flatten the eye. After the incisions heal, the cornea's shape is changed to offer better vision. The major disadvantage is that it permanently weakens the cornea and makes it more susceptible to trauma. Also, there is a much greater chance for scarring of the cornea that leads to discomfort [7]. Postoperative complications include halo, glare and astigmatism, and due to these complications and the advent of better techniques, RK is no longer very common [12].

Photo Refractive Keratectomy(PRK) uses an excimer laser to remove thin layers of cornea to sculpt a new contour of the corneal surface. However, this requires scraping the epithelium cells away because the laser cannot penetrate through that layer. The main advantage is that the cornea is not structurally weakened to any significant degree and the eye is not as vulnerable to trauma as in RK [7] [12].

Laser In Situ Keratomileusis (LASIK) is considered to be a simple operation that takes less than half of an hour to complete. It is a procedure that permanently changes the shape of the cornea, using an excimer laser. A surgical tool, called a microkeratome, is used to cut a flap in the cornea. A hinge is left at one end of this flap and the flap is folded back revealing stroma, the middle section of the cornea. 193 nm argon-fluoride excimer laser is used to ablate the stromal bed and the flap is pulled back to the original position to cover

the stroma after the laser usage [4, 19]. LASIK is proven to be effective in correction of high myopia, more accurate for myopia up to 12 D [18, 20]. The most frustrating complication of refractive surgery , in terms of visual acuity, is spherical aberration, because this increases the diffraction of the light rays and reduces contrast sensitivity for large pupils [5].

2.4 Optics

According to the wave theory of light, a point light source in air will emit light waves of identical speed in all directions, but as the wave travels through different media, the speed of light changes depending on the refractive indices of materials. These speed changes cause the wavefronts to bend or to refract. A lens is designed to bend wavefronts in a predictable manner, converting a parallel wavefront into a converging wavefront. The amount of refraction depends on each material's index of refraction and is determined by Snell's Law.

$$n_1 \sin\theta_1 = n_2 \sin\theta_2 \tag{2.1}$$

The variables n_1 and n_2 represent the indices of refraction and θ_1 and θ_2 represent the angles of refraction from the normal plane of the refractive surface. Figure 3.3 depicts the variables used in the formula. In optics, the refractive power of lens is represented by a unit called diopter. It is the reciprocal of the lens' focal length and denotes the degree to which it changes the refraction of light rays. For example, a lens capable of focusing parallel light rays 2 meters from the lens is said to have a focal length of 2 meters and its diopter measurement is 0.5. This measurement is important in our system especially when

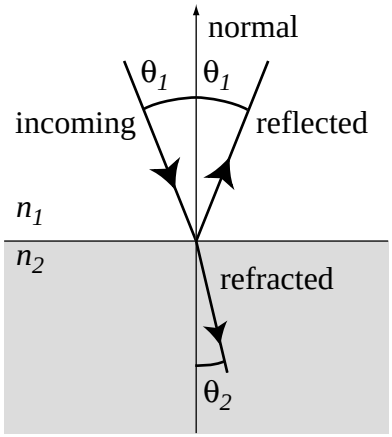


Figure 2.3: Snell's Law. Reprinted with permission from [8].

we discuss the implication of depth in the algorithm.

2.5 Shack-Hartmann Data

The Hartmann test was first performed in 1900 to test telescope optics by using a steel plate with holes drilled in a predefined pattern. In 1970, Ben Platt and Roland Shack expanded the concept to develop the Shack-Harmann sensor on a classified laser project for the U.S. Air Force. Since that time, the Shack-Hartmann sensor has been used by various organizations primarily for adaptive optics and wavefront aberration measurement applications in telescopes, lasers and optical systems. An array of microscopic lenses is the most important element of the Shack-Hartmann sensor. This lenslet array dissects the incoming light into a large number of apertures, and then measures the wavefront slope across each aperture. The sensor information is used to analyze the optical properties of the system that created the wavefront (in this case, the human eye) [22].

The Shack-Hartmann Sensor is a device that enables the precise measurement of

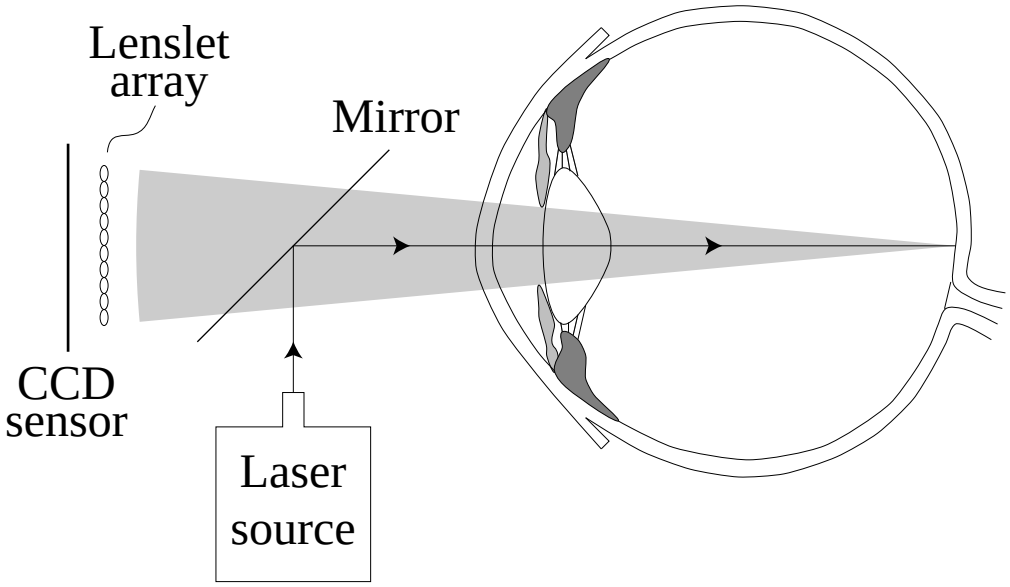


Figure 2.4: Shack-Hartmann Device. Reprinted with permission from [8].

the wavefront aberrations of an eye [13]. It is believed to be the most effective tool for the measurement of human eye aberration. A low-power laser beam is directed at the retina of the eye by means of a half-silvered mirror as in Figure 2.5. The retinal image of that laser now serves as a point source of light for a wavefront that passes through the eye's internal optical structures, past the pupil and eventually out of the eye. Finally, it is recorded on the CCD sensor placed in front of the eye [21, 23].

2.6 Point Spread Function (PSF)

The point spread function (PSF) is a critical structure in the algorithm because it determines the blurriness of the image. The PSF, also referred to as a *filter* or a *kernel*, describes how a ray of light is dispersed in a given space [8]. It is represented by a two-dimensional array and, as shown in Figure 2.6, resembles a 3-D histogram when plotted.

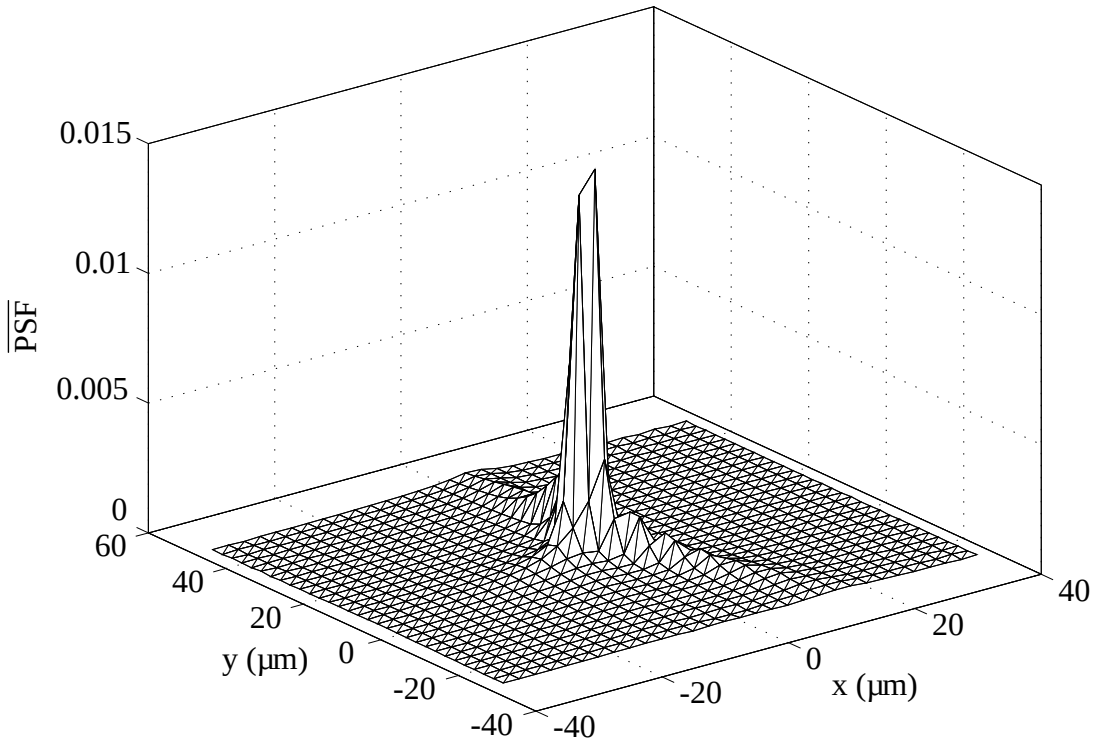


Figure 2.5: A sample PSF. Reprinted with permission from [8].

It provides information about predicted acuity in vision. To calculate the PSF for an eye, light rays are traced to simulate passing through the actual internal structures and onto the retina. The PSF is then used as a 2-D blur filter to be convolved with a pinhole image to produce the final blurred image. Since the optics of the eye do not change the number of photons, the integral of the PSF must equal unity. If the integral is less than unity, the final image will lose some of its initial energy, leading to a decrease in overall brightness. In our algorithm, we use a set of PSF's to blur objects located at different depths.

There are two different ways of constructing the PSF. The first approach is referred to as the traditional method, which is characterized by having the light source outside the eye and using the retina as the collection array. The collection array in this method is

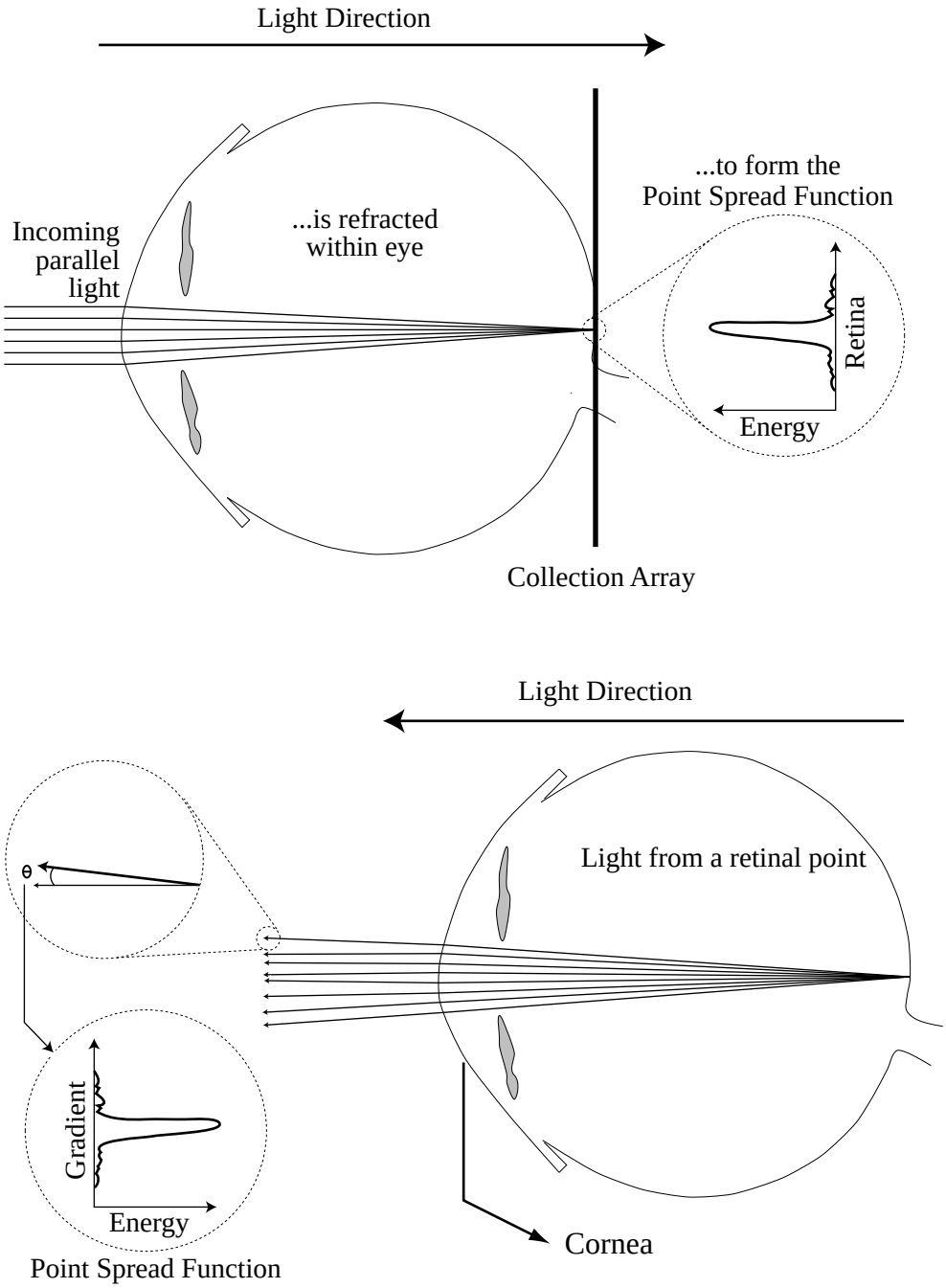


Figure 2.6: Traditional method of calculating PSF is at the top and Object Space method of calculating PSF (OSPSF) is at the bottom.

spatially indexed, meaning that the light rays' x and y coordinates of the landing sites are used to choose the cell of the array to increment.

We call the second approach an "object space" method. In this approach, the light source is located on the retina and the emerging rays are bent by the eye's optics. The collection array for this method is indexed using the gradient, which means that the angles of deviation in the x and y directions are used to choose the cell of the array. For a perfect eye, the gradient would be zero. The differences between these methods are depicted in Figure 2.3.

Chapter 3

Methods

3.1 Conceptual Overview

Shack Hartmann

The algorithm is designed to introduce perceptible distortions according to a specific human visual system. A Shack-Hartmann device is used to investigate the comprehensive behavior of light rays inside the eye, which are influenced by corneal deformity as well as other anatomical anomalies. As useful as the Shack-Hartmann data is, its low sampling rate makes it impossible to calculate an efficient eye model. Therefore, an extra process is required to increase the sampling rate by creating the continuous aberration data.

Re-sampling Wavefront

The data collected from the Shack-Hartmann device must be re-sampled in order to provide a high-density wavefront. A best-fit curve is derived by linear regression to include all the wavefront samples present in the Shack-Hartmann data. Calculating the PSF

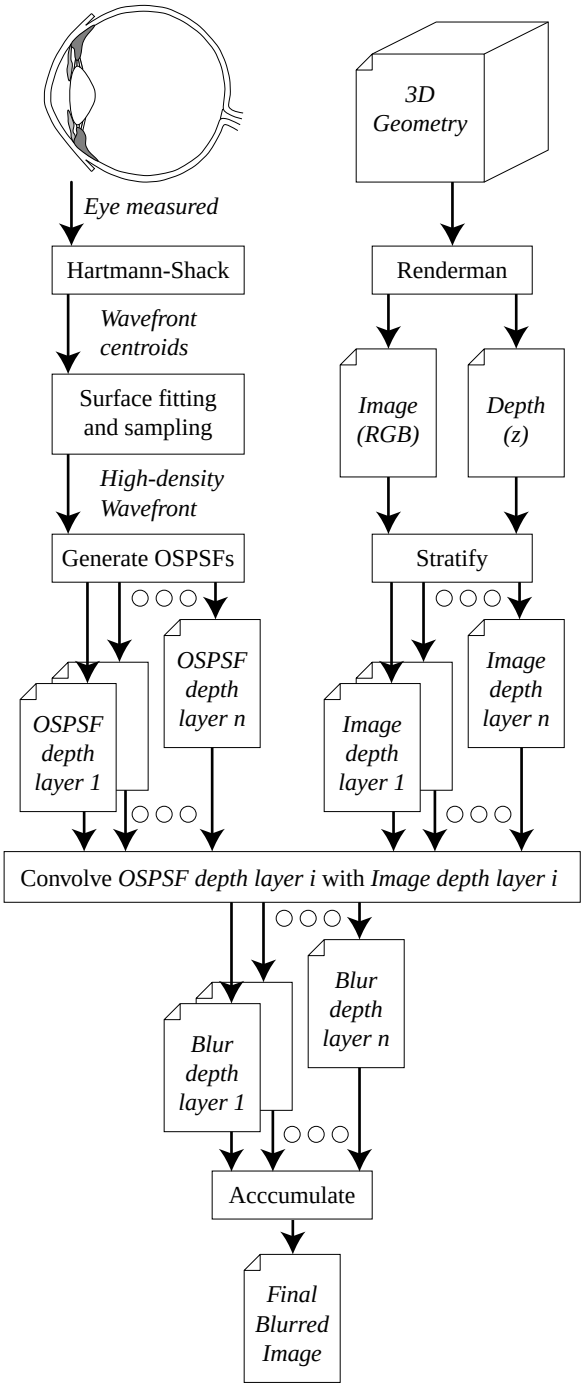


Figure 3.1: Overview of Algorithm. Reprinted with permission from [3].

simulates parallel light rays distributed at a much higher density than that of the Shack-Hartmann data. There are close to 80 samples in the Shack-Hartmann output whereas 300,000 samples are needed for the calculation. Re-sampling is performed by converting discrete values to a continuous medium represented by the best-fit curve and re-discretizing them by extracting aberration information at a much higher density.

PSF Construction

The next phase of the algorithm converts the wavefront information to PSF's, which act as blurring filters to the original image. Our algorithm uses the object space method to calculate the PSF (that is, the OSPSF) and it utilizes the slope rather than the position of the light ray. Once the slopes of the light rays are acquired from the re-sampled wavefront, a depth coefficient is applied to these slopes and the results are used to locate the cells to increment. The depth coefficient corresponds to the depth at which the current OSPSF is calculated. After the calculation, the array is normalized to ensure that the integral of the OSPSF equals unity and then the array is isolated as the final OSPSF.

The algorithm repeats the above step to compute the OSPSF for each discrete depth value. For each iteration, a different depth coefficient is used to produce the OSPSF for the particular depth, whereas the same set of re-sampled slope data is used. A set of OSPSF's is used to blur a single image. The resulting set of OSPSF's represents one eye, focused at a specific distance. Different sets of OSPSF's need to be calculated for different focusing distances, even if the same eye is simulated.

The depth values can range from zero to infinity, but they need to be converted to discrete values. The conversion is performed to exploit the pre-computed OSPSF's and to

facilitate separating the pixels of the original image according to the depths. The issue of preventing discontinuity in the final image is addressed by using small enough increments (in diopters) between the depths. This produces a continuous image that provides a smooth transition between pixels of different depths. Pixels of the original image are separated according to their depth values.

Convolution

The concept of having different PSF's for objects at different depths corresponds to the idea of having multiple filters for a single image. Besides the kind of eye that we are simulating, another factor considered for determining the appropriate set of blurring filters is where the eye is actually focused. The PSF at the depth where the eye is focused would produce a sharp image relative to the other PSF's. Actual blurring can be done by FFT or manual convolution. Finally, these separated, disjoint, blurred images are combined together like a jigsaw puzzle to produce a final image.

Given the distance of focus as a parameter, the system chooses the set of PSF's with which to perform convolution. The process accepts as inputs the separated image layers and the chosen set of PSF's, and generates a series of blurred images, with each blurred image corresponding to a separate layer. The algorithm uses the Fast Fourier Transform method to convolve each layer of image with the PSF of the corresponding depth. Combining these blurred layers forms the final image that simulates the vision through the specific human visual system.

3.2 Shack-Hartmann

As introduced in Section 2.5, the Shack-Hartmann device captures the wavefront information of the visual system as the laser beams exit the eye, passing through the cornea to reach the lenslets of the device. These lenslets are a collection of small lenses that are capable of focusing the wavefront on the video sensor. As expressed in equation 3.1, the Shack-Hartmann data at x and y coordinates comprise the slope at the same location multiplied by the focal length of the Shack-Hartmann device.

$$SH[x, y] = \nabla W[x, y]k \tag{3.1}$$

The aberration data is determined by the displacement of the deviation point from the reference point, which is the sub-image formed by the plane wavefront. In Equation 3.1, x and y are the corresponding coordinates in the device and k is a constant for converting wavefront to Shack-Hartmann values. In Figure 3.2, the green dots indicate the reference points and the red dots indicate the sub-image formed by the deviated wavefront. For the sample of an ideal eye with perfect vision, the green dots would coincide with the red dots. As can be seen in Figure 3.2, deviations of the red dots from the green dots (centers) are apparent in the Shack-Hartmann data of a LASIK patient. Image processing is employed to determine the center of each square and the deviation point (red dot) to sub-pixel resolution. Table 3.1 shows how the x and y coordinates of the red dots and the green dots are recorded in an output file. The wavefront slope is given by the differences between the respective coordinates of the red and green dots divided by the Shack-Hartmann lenslet focal length.

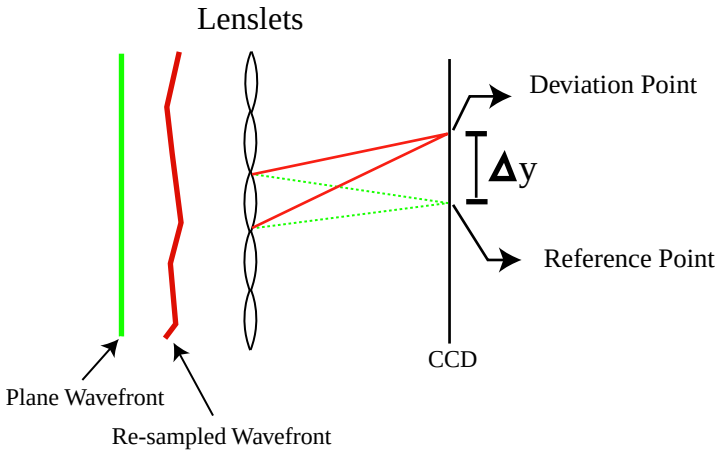


Figure 3.2: Measurement of slopes in the lenslets of Shack-Hartmann device

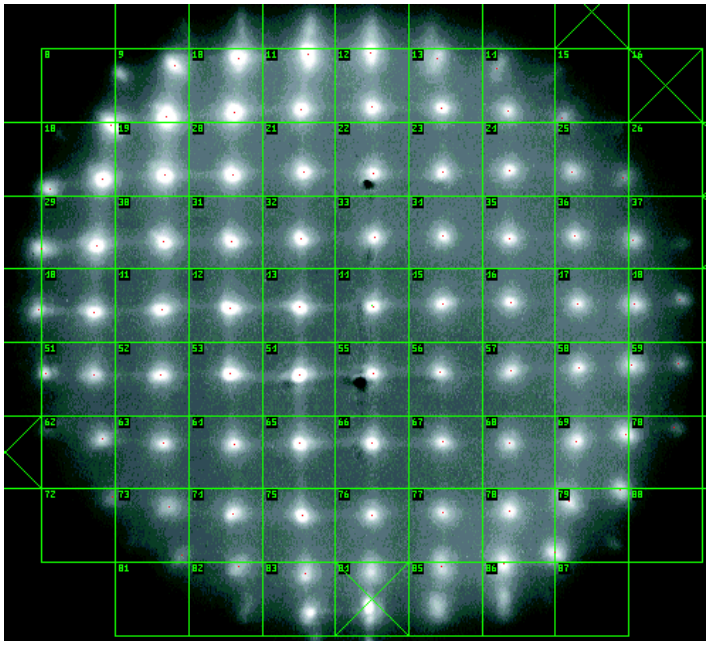


Figure 3.3: Actual output of a Shack-Hartmann device for a sample LASIK refractive surgery patient.

	Green dot x coordinate	Green dot y coordinate	Red dot x coordinate	Red dot y coordinate
1	278	126	284	146
2	329.5	126	331	137
3	381	126	382	133
4	433	126	431	135
5	484.5	126	476	140
6	226	178	238	198
7	278	178	278	187
8	329.5	178	329	184
⋮	⋮	⋮	⋮	⋮
83	433	540	426	519

Table 3.1: File of the x and y coordinates of the red and green dots generated from the Shack-Hartmann data shown in Figure 3.2.

3.3 Wavefront

The limited number of lenslets provides only a sparse sampling of the wavefront and consequently, the wavefront needs to be re-sampled at a much higher density. This can be accomplished by fitting the sparse Shack-Hartmann data with a continuous polynomial expansion

$$W(x,y) = \sum_{1 \le i+j \le 7} \mathbf{a}_{ij} x^i y^j \tag{3.2}$$

where a_{ij} are the Taylor's series coefficients. We constrain the sum of i and j to be between 1 and 7, which produces no more than 35 coefficients. Taking the derivatives of the expansion with respect to the variables x and y yields the following equations 3.3 and 3.4:

$$\frac{\delta W}{\delta x} = \sum_{1 \le i+j \le 7} \mathbf{a}_{ij} \mathbf{i} x^{i-1} y^j \tag{3.3}$$

$$\frac{\delta W}{\delta y} = \sum_{1 \le i+j \le 7} \mathbf{a}_{ij} \mathbf{j} x^i y^{j-1} \tag{3.4}$$

Linear regression is used to acquire the coefficients of the fitted curve and these derived equations are placed in a matrix as shown below in Equation 3.5 for a pseudo–inverse operation.

$$M_0 = \begin{bmatrix} \frac{\delta W}{\delta x} \\ \\ \frac{\delta W}{\delta y} \end{bmatrix} = \begin{bmatrix} ix^{i-1}y^j \\ \\ jx^i y^{j-1} \end{bmatrix} 1 \leq i + j \leq 7 \tag{3.5}$$

Variables i and j represent all possible integer pairs whose sum equals 1 through 7 iteratively and each column of the matrix M_0 represents each pair corresponding to a coefficient. The variables x and y represent the coordinates of every reference point (green dot) in the Shack-Hartmann data and each row represents a sample point. The size of M_0 is twice the number of samples by 35 because the calculation needs to be performed for both x and y directions. For the data of Table 4.1, the matrix would have 2 columns by 83 rows.

$$S = \begin{bmatrix} \Delta x \\ \\ \Delta y \end{bmatrix} \tag{3.6}$$

S is the matrix that contains x and y displacements of the sub-image (red dots) from the reference point (green dots). It is a single column matrix whose number of rows is twice the sample size. Using a pseudo–inverse operation on matrices M_0 and S, coefficients are calculated in a single column matrix with 37 rows. To re-sample the wavefront at a much higher rate, it is necessary to construct a matrix identical to M_0 in Equation 3.5.

$$M_1 = \begin{bmatrix} i{x_h}^{i-1}{y_h}^j \\ \\ j{x_h}^i{y_h}^{j-1} \end{bmatrix} 1 \leq i + j \leq 7 \tag{3.7}$$

However, M_1 needs to incorporate 300,000 samples by using the x and y coordinates of each location where re-sampling would occur. The variables x_h and y_h represent the 300,000

points where sampling would occur and the size of M_1 is 600,000 rows by 37 columns.

$$W = M_1 \times A \tag{3.8}$$

Matrix multiplication of the re-sampling matrix M_1 times the coefficient matrix A results in a single column matrix with 600,000 rows that represent the re-sampled wavefront. The first 300,000 values are the slopes in the x direction and the other half are slopes in the y direction.

The number of rays to be simulated is arbitrary because this is implemented as a parameter to the system. After experimentation, we decided that simulating 300,000 light rays would provide an adequate result without using too much system resources. The re-sampled data is interpreted in software as a huge two-dimensional array, similar to the way the PSF is depicted. Each element is a value corresponding to the slope of deviation that occurs at the specified location. For a perfect eye without any aberrations, the slope at every point on the wavefront would have a zero value. This process generates an array that contains 300,000 lines, with each line comprising four elements: x-coordinate, y-coordinate, the slope in x direction, and the slope in y direction.

3.4 PSF Construction

Our algorithm uses the object space method (OSPSF) to compute the PSF (OS-PSF). The angle (measured in minutes, that is 1/60 degree) subtended by each pixel is a parameter to the algorithm to indicate the allocation of the real world space. Having 0.5 minute per pixel yields an image that represents a small segment of the view. For example, if the size of a given image is 512 by 512 pixels, the entire image covers $512 \times 0.5 \times \frac{1}{60} = 4.26$

	X Coordinate	Y Coordinate	ΔX	ΔY
1	-3.5000	-3.5000	13.1843	8.5034
2	-3.5000	-3.4872	13.0880	8.4142
3	-3.5000	-3.4744	12.9924	8.3259
4	-3.5000	-3.4617	12.8977	8.2384
5	-3.5000	-3.4489	12.8037	8.1518
6	-3.5000	-3.4361	12.7104	8.0661
7	-3.5000	-3.4233	12.6179	7.9812
8	-3.5000	-3.4105	12.5261	7.8971
⋮	⋮	⋮	⋮	⋮
⋮	3.4908	3.4908	3.0233	1.2970

Table 3.2: Re-sampled wavefront for LASIK eye.

degree by 4.26 degree region in the field of vision. This zooming effect causes the PSF to cover a small area and the collection array takes dimensions of 255 by 255. On the other hand, if we were to increase the angle subtended by each pixel to be 8 minutes per pixel, then the same image would represent the world 16 times larger in each direction. The same image would then cover a 68.27 by 68.27 degree field of vision. Since the world it represents is much larger, the blur size decreases reciprocally. Its PSFs occupy less real-world space, but the collection array is one-quarter of the size (127 by 127 in pixels) because the peaks and tails of the histogram are localized within a small region at the center, and having more pixels would only waste system resources. The pupil size is another parameter in the algorithm and this simulates the human pupil by controlling the number of light rays entering the eye. Typically, a real world image would be convolved using the 8-minutes-per-pixel format and the 127-by-127 pixel PSF's. The 0.5-minute-per-pixel format is used in test scenarios where the blurring effect needs to be exaggerated.

A particular PSF that is representing the layer that is at the focused depth would

	X Coordinate	Y Coordinate	ΔX	ΔY
1	-3.5000	-3.5000	0	0
2	-3.5000	-3.4872	0	0
3	-3.5000	-3.4744	0	0
4	-3.5000	-3.4617	0	0
5	-3.5000	-3.4489	0	0
6	-3.5000	-3.4361	0	0
7	-3.5000	-3.4233	0	0
8	-3.5000	-3.4105	0	0
⋮	⋮	⋮	⋮	⋮
⋮	3.4908	3.4908	0	0

Table 3.3: Re-sampled wavefront for ideal eye.

have a center peak without any tail component given that the person has perfect vision. The next PSF in the set, which corresponds to a depth farther than of the first PSF, would have a shorter peak with a small tail component. The farther away a PSF is from the focused depth, the flatter it becomes, leading to a more distorted image (shown in Figure 3.4).

The final step of the PSF calculation is the process of normalization. The sum of the values in a PSF array should be unity in order to maintain the same brightness as that of the original image. The algorithm normalizes the resultant PSF by dividing each element in the array by the sum of the array elements.

$$finalPSF(x,y) = \frac{PSF(x,y)}{\sum PSF} \tag{3.9}$$

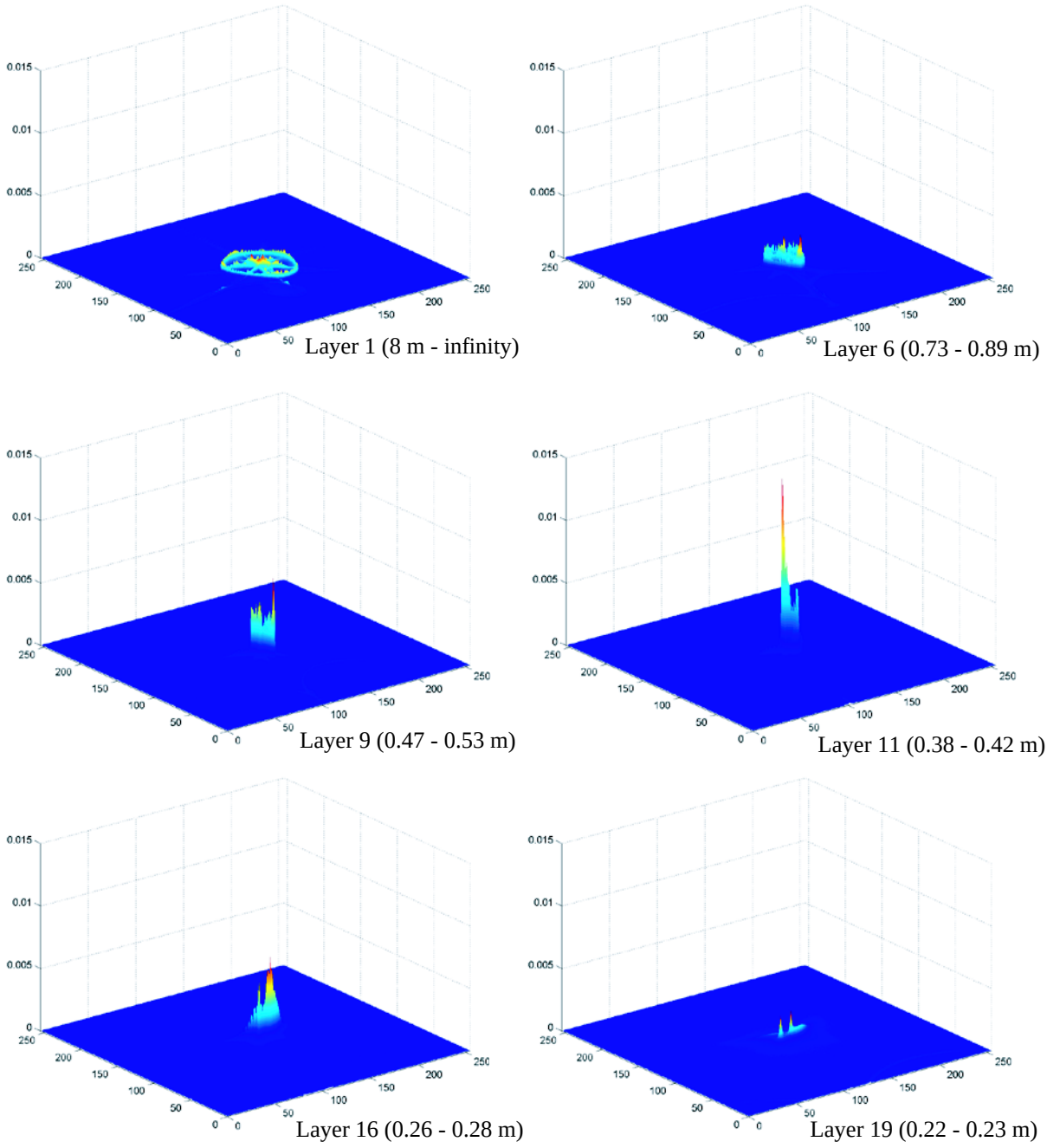


Figure 3.4: PSF's for LASIK eye focused at 0.4 m. The Z axis is fixed at 0.015. Six of the forty-one PSF's are shown and corresponding depth for each histogram is in the parenthesis.

3.5 Depth Values and Stratification

The discretization of the continuous depth values introduces a patchy final image if the spacing of the discrete values is too wide. Based on the limits of human visual acuity, the range of depths is from 0 to 10 diopters and the spacing between adjacent layers is 0.25 diopter in order to provide smooth transition. Since the increment between each layer is 0.25 diopters, this range of depth produces 41 layers of PSF's. Note that 0 diopter represents an infinite distance ($\frac{1}{\infty}$ m = 0 diopter) and 10 diopters represents a distance of 10 cm ($\frac{1}{0.1}$ m = 10 diopters). It is interesting that all the objects located farther than 8 ($\frac{1}{0.25}$ diopters = 8 m) meters are blurred with the PSF corresponding to infinity and that everything located closer than 10 cm is blurred with one corresponding to 10 cm.

The dimensions of the depth array is the same as the dimensions of the image, except that each element contains one distinct value of depth instead of the three values that would represent red, green, and blue values in the image array. The depth information indicates the distance from the viewpoint for the corresponding pixel and is determined and saved from the 3-D rendering software that produced the original image. The depth information needs to be converted into discrete values. This is obtained by the following formula:

$$DiscreteDepthIndex = ROUND(\frac{1}{ContinuousDepth} \times 4) + 1 \tag{3.10}$$

According to the formula, the continuous depth of 0.1 meter will correspond to the discrete depth index of 41, and the continuous depth of infinity will correspond to the discrete depth 1. After this step, pixels of the image will have been separated into different

Layer	Depth Range	Layer	Depth Range
1	$8.000000\text{m} < x \leq \text{Infinity}$	21	$0.195122\text{m} < x \leq 0.205128\text{m}$
2	$2.666667\text{m} < x \leq 8.000000\text{m}$	22	$0.186047\text{m} < x \leq 0.195122\text{m}$
3	$1.600000\text{m} < x \leq 2.666667\text{m}$	23	$0.177778\text{m} < x \leq 0.186047\text{m}$
4	$1.142857\text{m} < x \leq 1.600000\text{m}$	24	$0.170213\text{m} < x \leq 0.177778\text{m}$
5	$0.888889\text{m} < x \leq 1.142857\text{m}$	25	$0.163265\text{m} < x \leq 0.170213\text{m}$
6	$0.727273\text{m} < x \leq 0.888889\text{m}$	26	$0.156863\text{m} < x \leq 0.163265\text{m}$
7	$0.615385\text{m} < x \leq 0.727273\text{m}$	27	$0.150943\text{m} < x \leq 0.156863\text{m}$
8	$0.533333\text{m} < x \leq 0.615385\text{m}$	28	$0.145455\text{m} < x \leq 0.150943\text{m}$
9	$0.470588\text{m} < x \leq 0.533333\text{m}$	29	$0.140351\text{m} < x \leq 0.145455\text{m}$
10	$0.421053\text{m} < x \leq 0.470588\text{m}$	30	$0.135593\text{m} < x \leq 0.140351\text{m}$
11	$0.380952\text{m} < x \leq 0.421053\text{m}$	31	$0.131148\text{m} < x \leq 0.135593\text{m}$
12	$0.347826\text{m} < x \leq 0.380952\text{m}$	32	$0.126984\text{m} < x \leq 0.131148\text{m}$
13	$0.320000\text{m} < x \leq 0.347826\text{m}$	33	$0.123077\text{m} < x \leq 0.126984\text{m}$
14	$0.296296\text{m} < x \leq 0.320000\text{m}$	34	$0.119403\text{m} < x \leq 0.123077\text{m}$
15	$0.275862\text{m} < x \leq 0.296296\text{m}$	35	$0.115942\text{m} < x \leq 0.119403\text{m}$
16	$0.258065\text{m} < x \leq 0.275862\text{m}$	36	$0.112676\text{m} < x \leq 0.115942\text{m}$
17	$0.242424\text{m} < x \leq 0.258065\text{m}$	37	$0.109589\text{m} < x \leq 0.112676\text{m}$
18	$0.228571\text{m} < x \leq 0.242424\text{m}$	38	$0.106667\text{m} < x \leq 0.109589\text{m}$
19	$0.216216\text{m} < x \leq 0.228571\text{m}$	39	$0.103896\text{m} < x \leq 0.106667\text{m}$
20	$0.205128\text{m} < x \leq 0.216216\text{m}$	40	$0.101266\text{m} < x \leq 0.103896\text{m}$
		41	$0.098765\text{m} < x \leq 0.101266\text{m}$

Table 3.4: Depth Chart. Lists the depth range for each layer

layers according to their discrete depth indices. Thus, it is possible to have a maximum of 41 layers to represent one image. The algorithm loops through each image layer and chooses the corresponding PSF from the set that has been already calculated.

3.6 Convolution

Convolution is an image manipulation technique in which each pixel of an image has its value modified depending on values of its neighboring pixels. A convolution operation combines the colors of a source pixel and of its neighbors to determine the color of a destination pixel. This combination is specified using a filter (the PSF), which is an operator that specifies the proportion of each source pixel color should contribute to the destination pixel color. We can think of the filter as a template that is overlaid on the image to perform a convolution on one pixel at a time. As each pixel is convolved, the template is moved to the next pixel in the source image and then the convolution process is repeated. A source copy of the image is used for input values for the convolution, and all output values are saved into a destination copy of the image. Thus, the updated pixel values are not used as input for modifying adjacent pixels. The center of the filter can be thought of as overlaying the source pixel being convolved. In Figure 3.6, a convolution operation that uses the first filter has no effect on an image: each destination pixel has the same color as its corresponding source pixel, whereas the second filter generates a blurred image. The important rule for creating filters is that the elements should all sum to unity to preserve the brightness of the image [1]. Figure 3.6 depicts how convolution is performed.

In 2-D discrete space, at a specified depth:

$$c = a \otimes b_d = \sum_{j=-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} a[j,k] \cdot b_d[m-j, n-k] \tag{3.11}$$

where a denotes the original image, b_d denotes the filter at the specified depth, m denotes x coordinate within the filter, n denotes y coordinate within the filter, and c denotes the convolved image. It is important to notice that this equation is for one depth layer; to determine the final image, the summation over all the depths is needed.

3.6.1 Fast Fourier Transform(FFT) Convolution

The form of a convolution as explained in Section 3.6 is seldom used because of its extravagant time requirement. Looking up many neighboring values for each pixel in an image whose dimensions are 512 by 512 would be very slow.

The crucial importance of the Fast Fourier Transform (FFT) is that it converts convolution into a pointwise multiplication operation. It is much quicker to do a complex rotation (the FFT), then do a simple manipulation, and then rotate back (the inverse FFT) than it is to implement convolution directly. Using the Fast Fourier Transform is a more efficient way of calculating the final image. Instead of traversing the entire image array, the convolution is simplified to numerous matrix operations. The obvious advantage of the FFT convolution is the gain in speed. However, to take advantage of it, a square image with the dimension of each side being 2^n, where n is a positive integer, needs to be used as the original image.

A signal can be viewed from two different standpoints:

- Spatial Domain

Point Spread Function

0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	1.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

Point Spread Function

0.03	0.03	0.03	0.03	0.03
0.05	0.05	0.05	0.05	0.03
0.03	0.05	0.1	0.05	0.03
0.03	0.05	0.05	0.05	0.03
0.03	0.03	0.03	0.03	0.03

Figure 3.5: The PSF on the left has no effect on the original picture and produces a crisp image. All of the PSF's energy is localized at the center of the array. The PSF on the right has some smearing of the energy and the convolution results in a slightly blurred image.

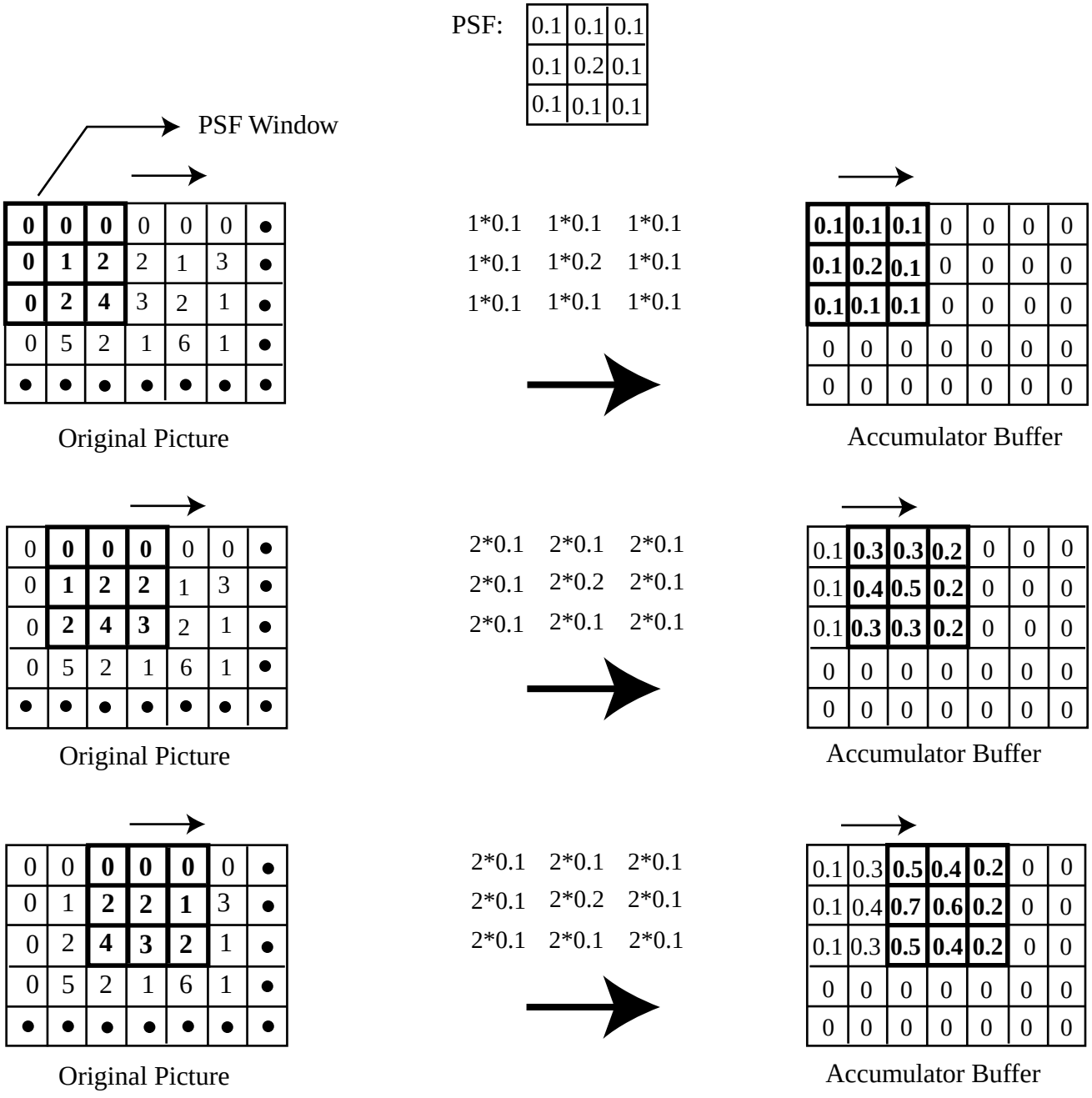


Figure 3.6: Process of Manual Convolution.

- Frequency Domain

An example of using *spatial domain* would be the trace on an oscilliscope where the vertical deflection is the signal's amplitude, and the horizontal deflection is the time variable. The second representation is the *frequency domain*. An example is the trace on a spectrum analyser, where the horizontal deflection is the frequency variable and the vertical deflection is the signal's amplitude at that frequency. Any given signal can be fully described in either of these domains. We can transform between the two by using a tool called the Fourier Transform. Depending on what we want to do with the signal, one domain tends to be simpler than the other. The equation for the Discrete Fourier Transform is complicated to work out as it involves many additions and multiplications involving complex numbers [1, 6]. The steps for the FFT convolution are the following:

1. Compute A(Ω,Ψ)=F(a[m,n])

2. Multiply A(Ω,Ψ) by the precomputed(Ω,Ψ)=F(h[m,n])

3. Computer the result c[m,n]=F^{-1}(A(Ω,Ψ)*(Ω,Ψ))

where *a* refers to the original image, and *h* refers to the filter (PSF). For each discrete depth value, a different PSF is chosen for convolution. The PSF is embedded at the center of an array whose dimensions are the same as the original image. The elements in this array are initially zero. Then, FFT shifting is performed to switch the first and third quadrants, and switch the second and fourth quadrants of the array. The two-dimensional Discrete Fourier Transform of the embedded PSF is multiplied with the Discrete Fourier Transform of the pixels isolated according to its depth. Taking the inverse FFT of the product produces the

blurred image for the corresponding depth. The process is depicted in Figure 3.6.1.

After each layer of the original image is convolved with the corresponding PSF, the final blurred image is formed by adding all the disjoint layers to a black image.

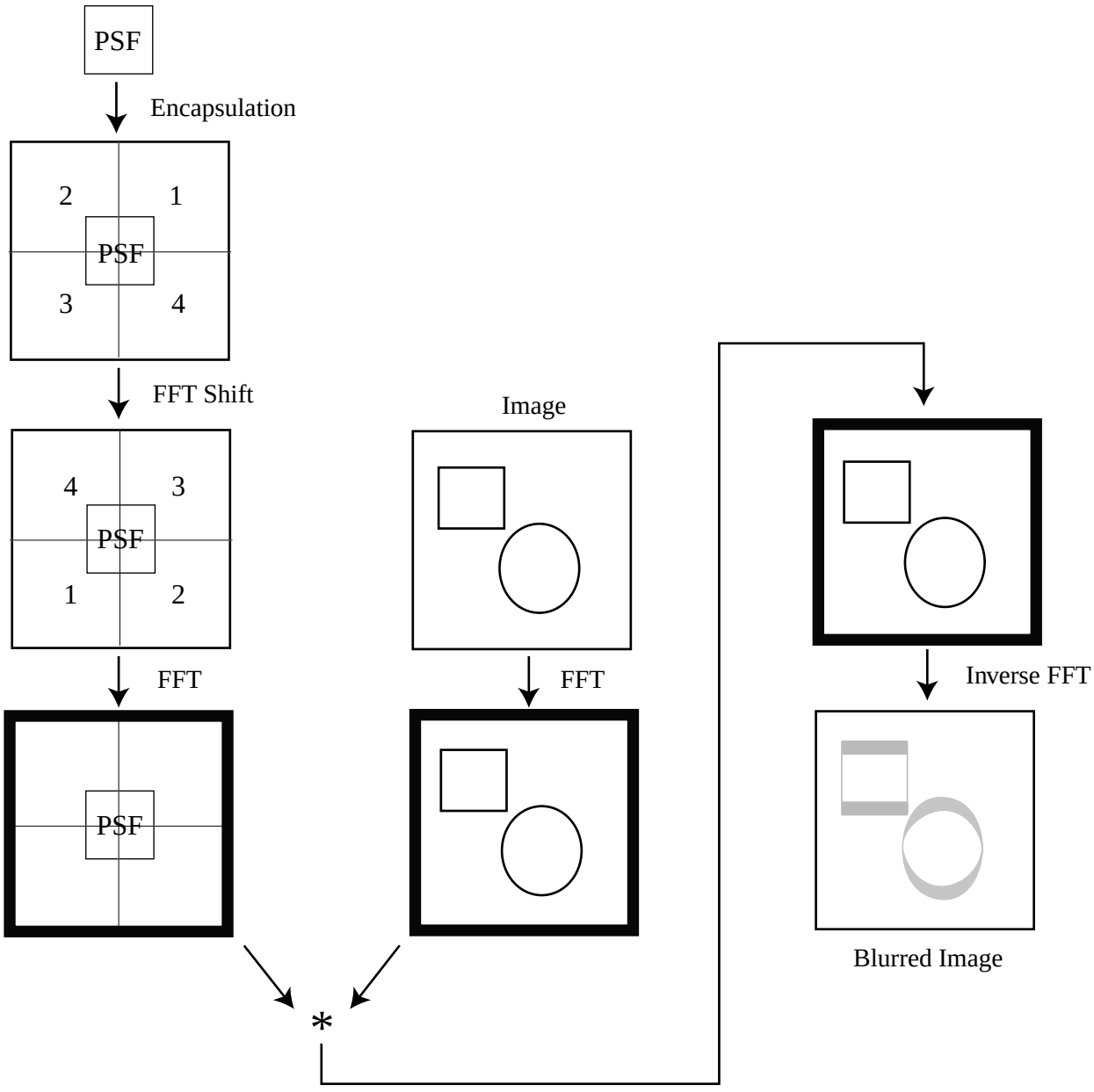


Figure 3.7: Process of FFT convolution

Chapter 4

Point Spread Functions

The generation of each set of point spread functions takes 20 to 30 minutes on a 500 Mhz Pentium III machine with 448 MBytes of main memory. This excessive requirement is caused by the simulation and calculation of $12,300,000$ ($300,000 rays/PSF * 41 PSF's = 12,300,000$) light rays. Ideal and astigmatic eyes are mathematically simulated, whereas the LASIK PSF's are actually derived from the Shack-Hartmann data of a patient who underwent the operation.

4.1 Pupil Size

Varying the pupil size of the eye changes the PSF shape. As the pupil size decreases, the PSF histogram gets more localized at the center and this results in crisper final images when used in convolution. The histograms shown in Figure 6.1 are calculated for an ideal eye focused at 0.2 meters as the pupil size increases from 0.5 mm to 4 mm. A separate calculation was required for each pupil size, with each resulting in forty one

different PSF's, but only the first layer of each set is displayed in the figure. The first layer is used to blur objects that are located 8 meters and beyond. The simulation is done in a 0.5 minutes-per-pixel world using 255-by-255 PSF dimensions.

4.2 Ideal Eye

For the case of an ideal eye, the pupil size is fixed at 4 mm and the focusing distance is 0.2 meters in the 0.5 minutes-per-pixel environment. The dimension of the PSF for the ideal eye is 255 by 255 pixels. Figures 4.2 and 4.3 show the PSF's created for the ideal eye. The first layer, corresponding to 8 meters and beyond, displays a flat histogram that encompasses a circular region in the middle. As the layer approaches the focusing depth, the encompassing area decreases, but this is compensated by the significant increase in the height. The twenty-first layer corresponds to the 0.195 - 0.205 m range and is characterized by a single peak that results in a clear image. Even though the depth difference between the twenty-first and twenty-second layer is only 0.25 diopter, there is a tremendous difference between the heights of the peaks. The twenty-first layer's histogram peaks at 1 and the twenty second layer's peaks at 0.03, which causes a significant distortion in the final image. As the depth gets closer to the viewer, the histogram becomes flatter and more spread out on the collection array.

4.3 Real LASIK Eye

For calculating the PSF's for the LASIK eye, the pupil size is fixed at 7 mm and the eye is focused at 0.2 meters in a 0.5 minutes-per-pixel world. The dimension of the PSF for

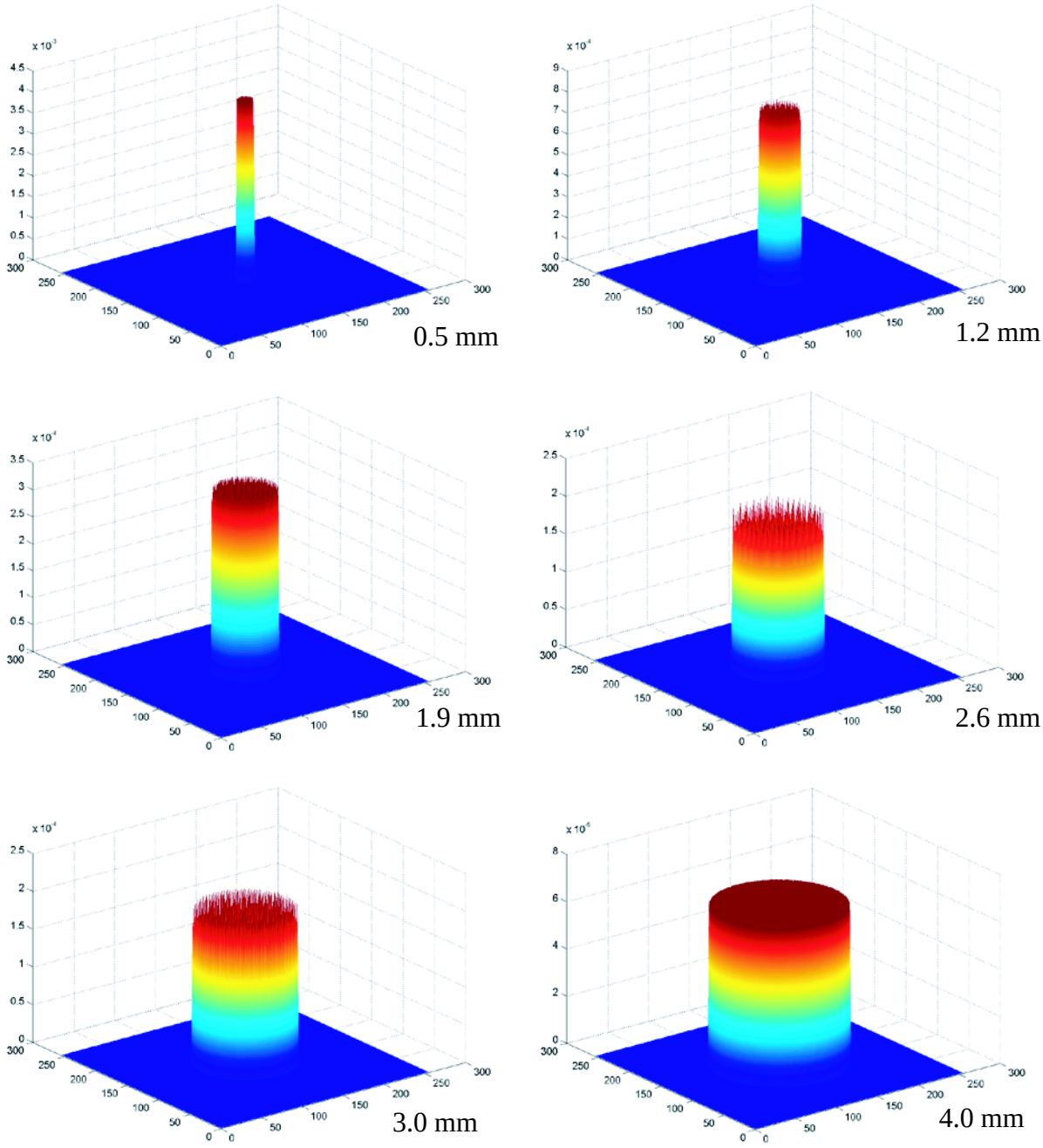


Figure 4.1: PSF. Ideal eye focused at 0.2 m. Layer 1 (for 8 m and beyond) of PSF's for the different pupil sizes. The pupil size is given as diameter in millimeter. The Z-axis is scaled automatically.

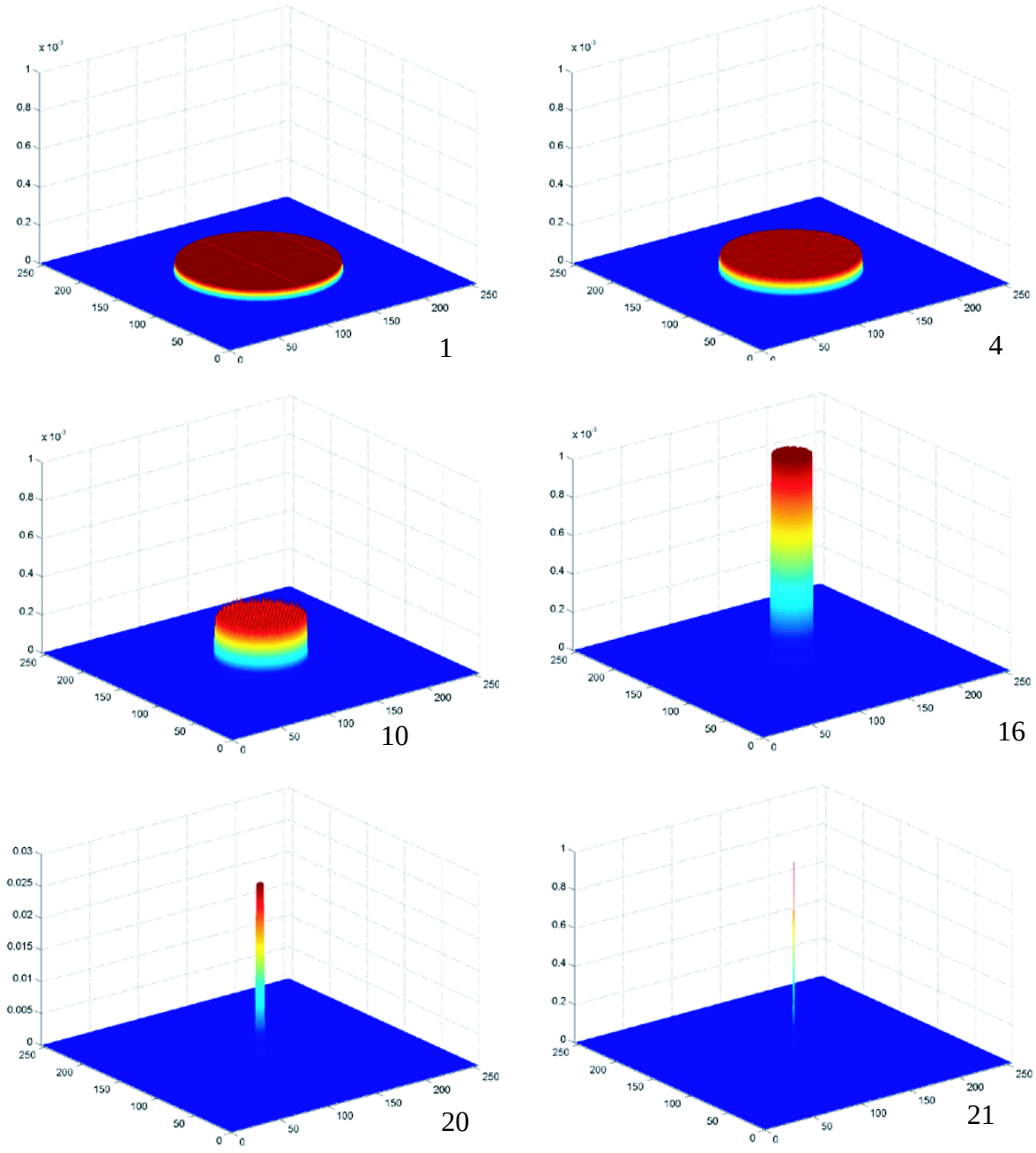


Figure 4.2: PSF. Ideal eye focused at 0.2 m. The value below each histogram is the depth layer number. For example, layer 1 is used to blur objects located 8 meters and beyond. The scale of Z axis has been changed to accommodate the values.

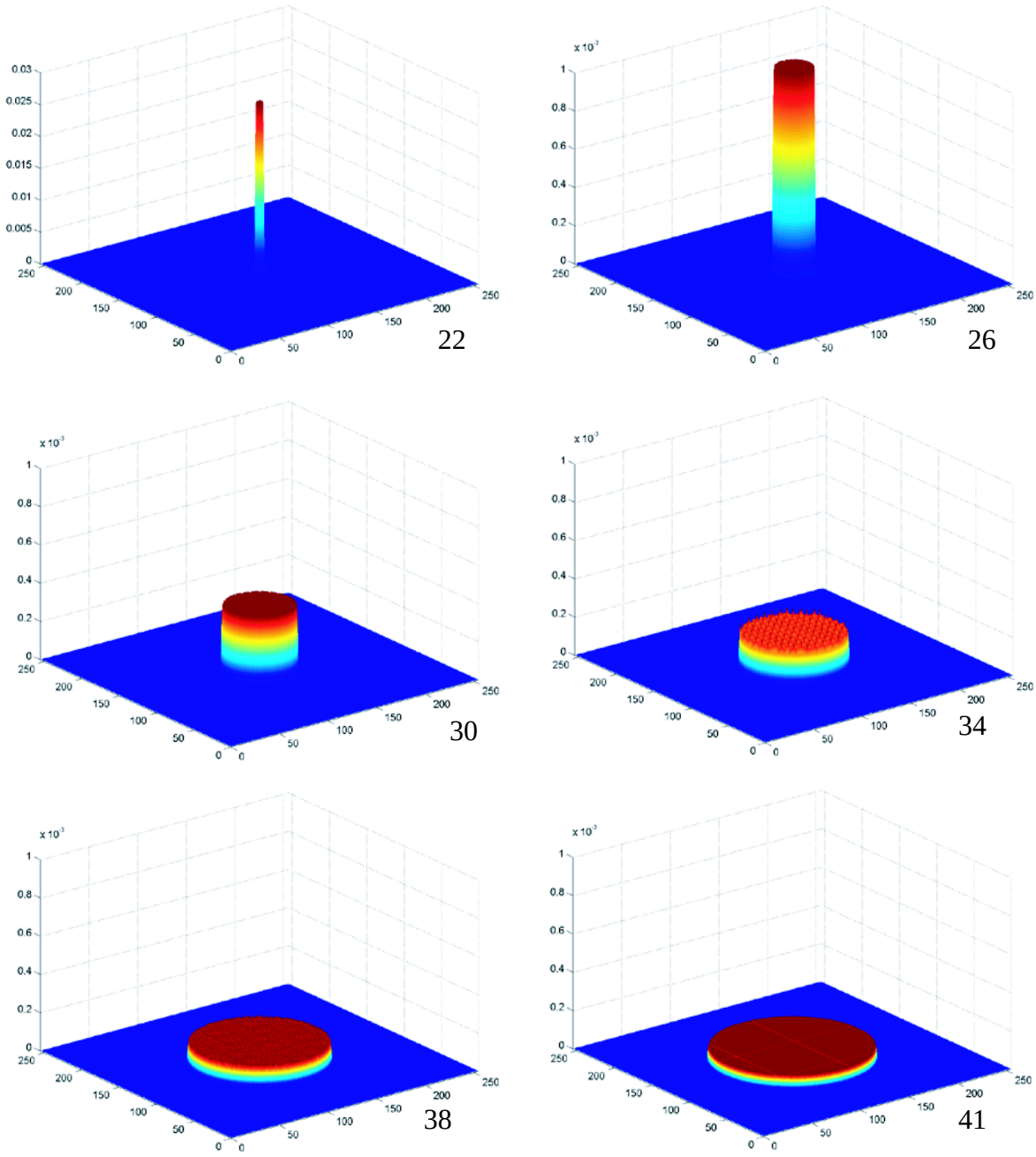


Figure 4.3: PSF. Ideal eye focused at 0.2 m. Layers 22,26,30,34,38,and 41.

the real LASIK eye is 255 by 255 pixels. As shown in Figures 4.4 and 4.5, the irregularity and asymmetry of the PSF's suggest that its data is much more realistic than the ideal eye. The twenty-first PSF still corresponds to the focusing distance and this PSF produces the clearest image relative to the others in the set. Distribution of energy is apparent. The z-axis is fixed at 0.015 radiance count unit which is only 1.5 % of the histogram for the twenty-first PSF of the ideal eye.

4.4 Astigmatic Eye Simulations

For the astigmatic simulation, the pupil size is fixed at 4 mm and the eye is focused at 0.2 meters away from the viewer in a 0.5 minutes-per-pixel environment. The dimension of the PSF for this simulation is 255 by 255 pixels. The simulations are done by keeping the dioptric power in the x direction to 0 and the power in the y direction to 2 diopters. The base of the PSF histogram, which was circular in the ideal case, is changed to an ellipse due to the accentuated aberration in one direction. In Figure 4.6, the 21st PSF that corresponds to the focusing distance is completely flattened in the x direction, but the aberration of 2 diopters is maintained in the y direction. In Figure 4.7, the 29th PSF, which corresponds to the depth range between 0.140 and 0.145 m, is 2 diopters away from the 21st PSF ($29 - 21 = 8; 8 * 0.5D = 2D$). The artificially introduced 2 diopter component is cancelled by the depth component of the 29th layer, which explains the flatness in the y direction. However, the 2 diopter depth component is still exhibited in the x direction.

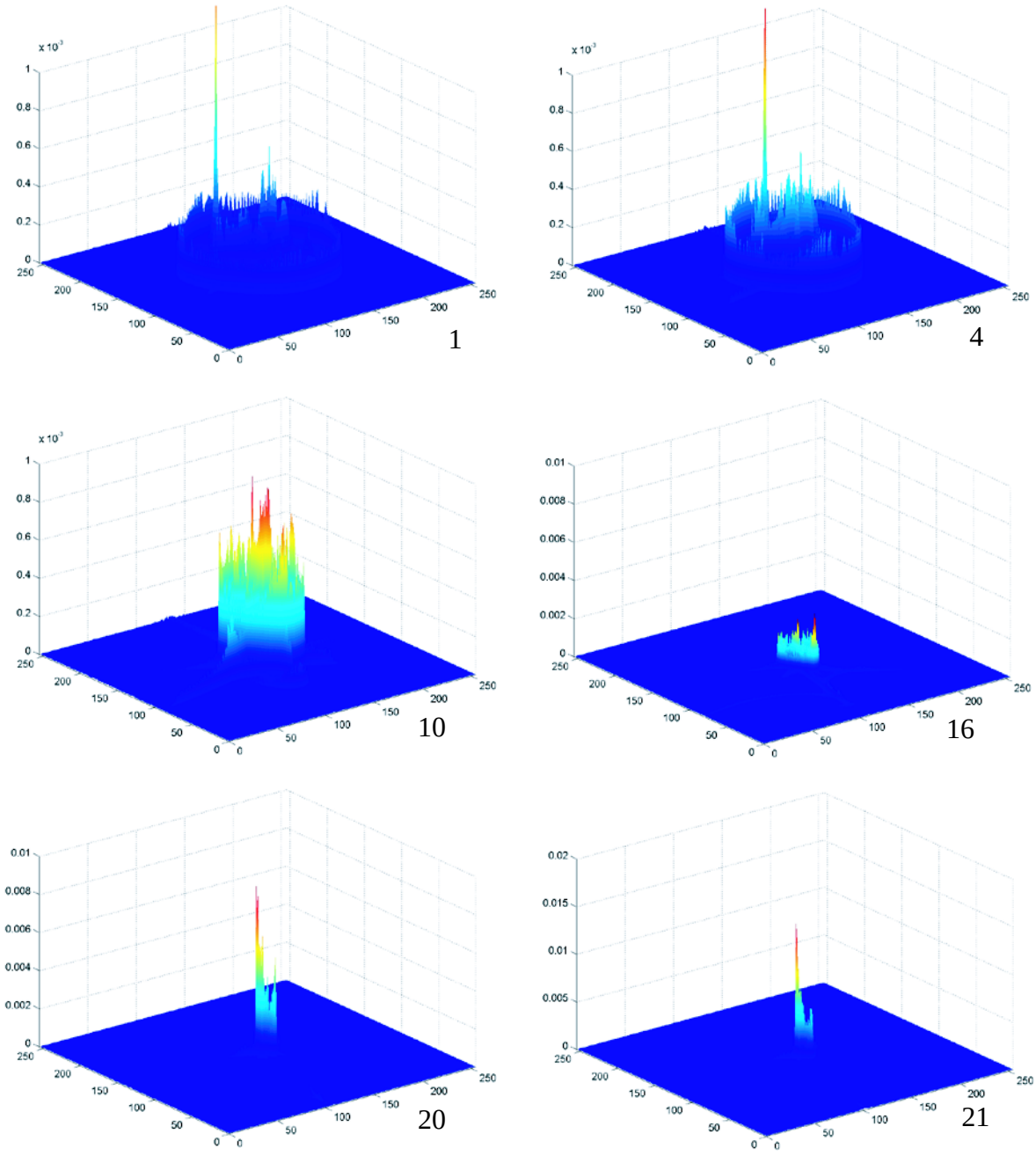


Figure 4.4: PSF. LASIK eye focused at 0.2 meters. Layers 1, 4, 10, 16, 20, and 21. The Z-axis has been auto-scaled.

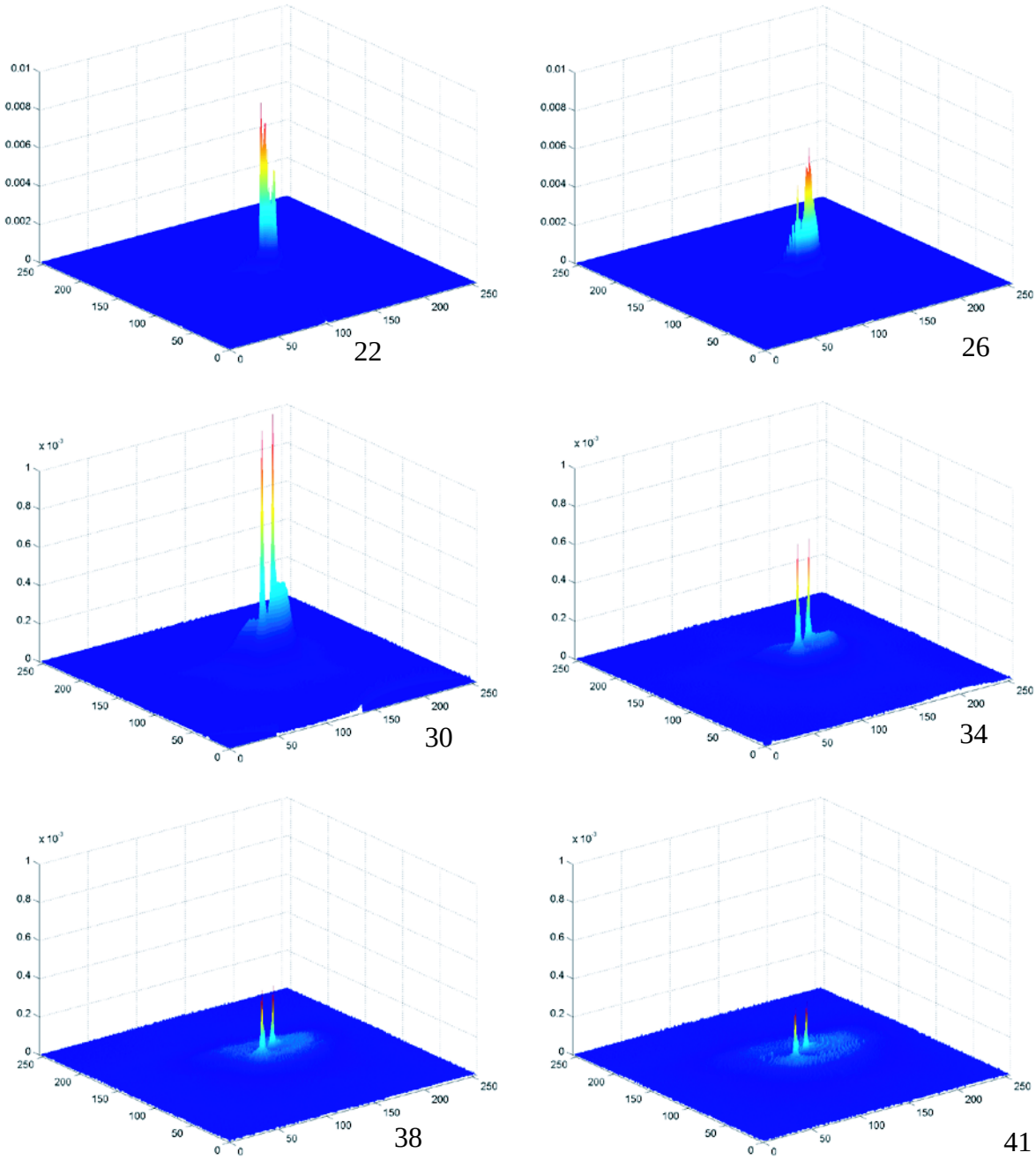


Figure 4.5: PSF. LASIK eye focused at 0.2 meters. Layers 22, 26, 30, 34, 38, and 41.

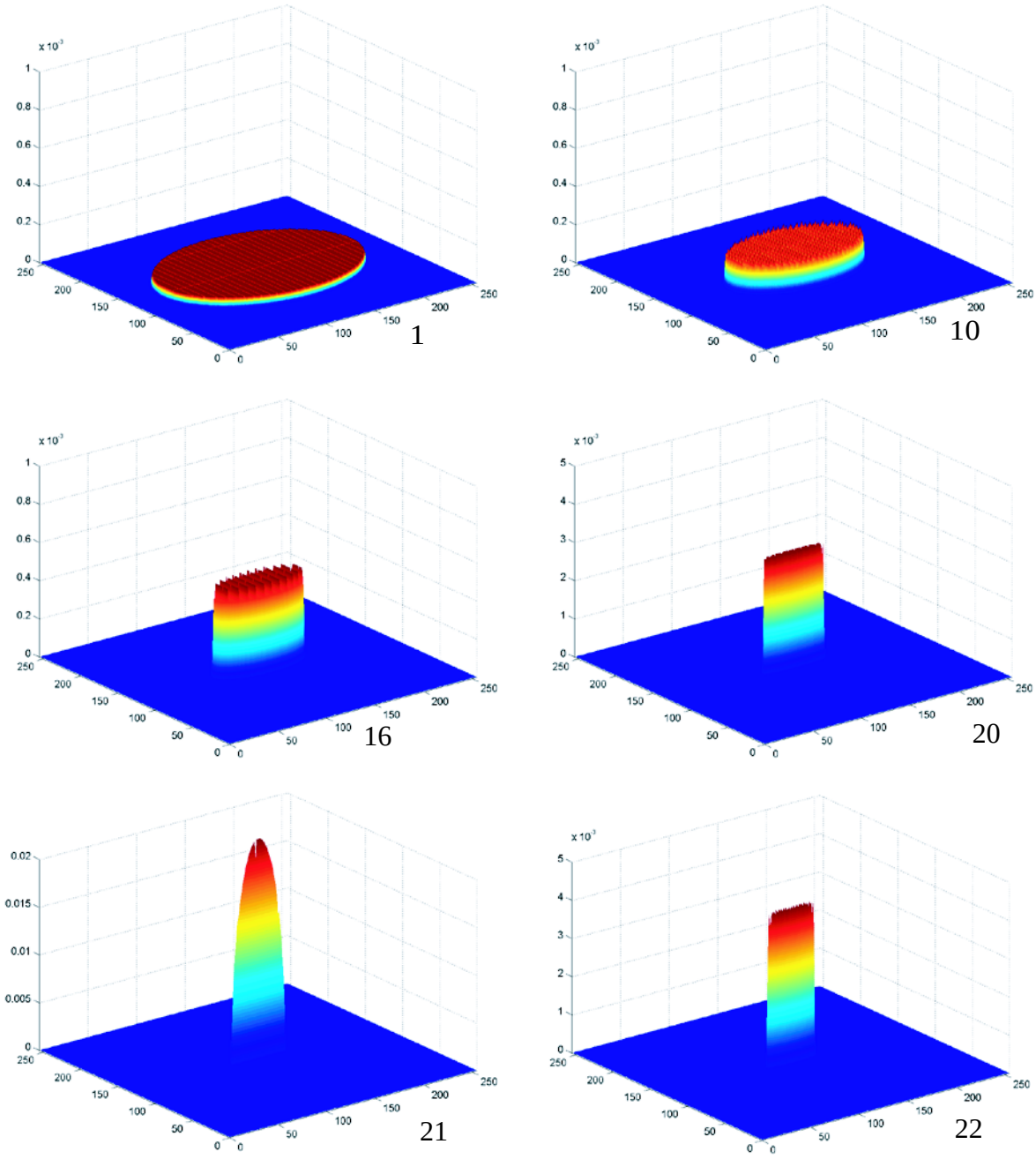


Figure 4.6: PSF. Layer 1, 10, 16, 20, 21, 22 of the PSF's for the astigmatic eye, focused at 0.2 meters. The scale of Z axis has been changed to accommodate the values.

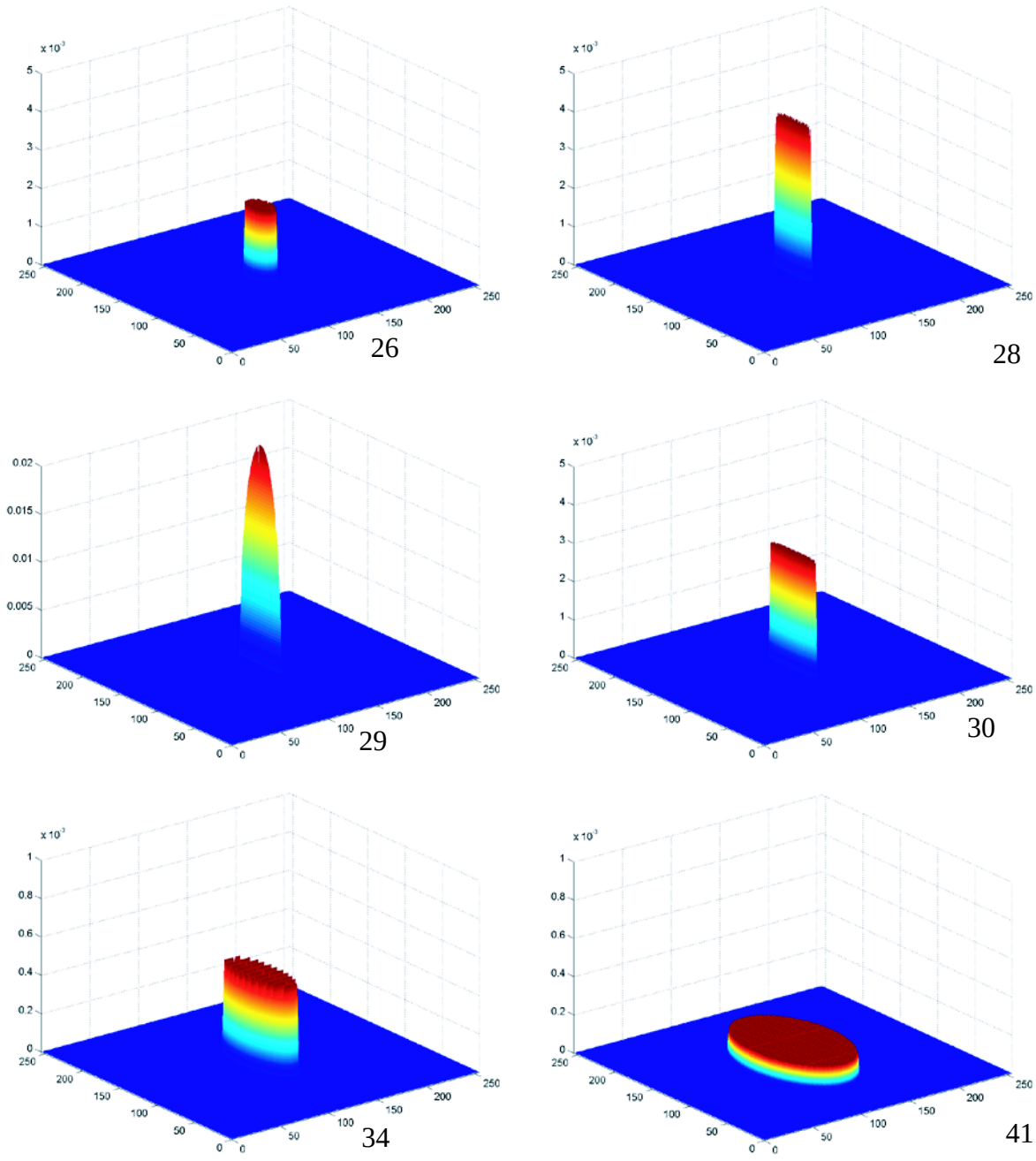


Figure 4.7: Layer 26, 28, 29, 30, 34, 41 of the PSF's for the astigmatic eye, focused at 0.2 meters. The scale of Z axis has been changed to accommodate the values.

Chapter 5

Simulations

The algorithm successfully generated four scenes("cubes", "road", "real life", "car") using the calculated PSF's. The TIFF images for all the scenes and the corresponding depth files were created using Renderman, a 3D ray-tracer/rederer. The Tree image was captured by a digital camera and the depth file was manually created. The FFT method was used to perform convolution.

5.1 Cubes Scene

The image resolution is 1024 by 1024 pixels. The white cube is located at 0.1 meters, the red cube at 0.2 meters, the green cube at 0.3 meters, and the blue cube at 0.4 meters away from the viewer. The image is set in the 0.5 minutes per pixel world, which means that every object in the scene actually is a miniature version of what it seems to be. This setting also implies that the whole image encompasses 8.53 by 8.53 (1024 pixels * 0.5 minutes * $\frac{1\ degree}{60\ minutes}$=8.53) degrees in the field of vision.

The faces of the focused cube are perfectly crisp in images seen through the ideal eye. For the astigmatic eye (Figure 5.4), the image whose focal depth is 0.3 m shows crisp vertical edges and distorted horizontal edges on the green cube. The red cube which is located 0.2 m away only has its vertical edges distorted because the difference in diopters is close enough to 2 D ($\frac{1}{0.2} - \frac{1}{0.3} = 1.6D$). The LASIK eye shows its random distortion caused by the real eye, opposed to the mathematically simulated ones.

5.2 Road Scene

The image resolution is 1024 by 1024 pixels. Starting from the closest sign on the right side of the road to the farthest, signs are located at 0.056, 0.11, 0.22, 0.45, 0.89, 1.8 and 3.9 meters away from the viewer. The image uses the 0.5 minutes per pixel environment, which implies that the scene actually encompasses a very small region (8.35 by 8.35 degree) in the field of view.

Figure 5.5 shows the original image seen through a pinhole camera model. The images in Figures 5.6, 5.7, and 5.8 are focused at different signs of the road and blurs are apparent on the signs that are not in focus. The blurs may seem excessive and exaggerated, but keep in mind that it represents an image of a tiny road because of the usage of 0.5 minutes per pixel. When the eye is focused at an infinite distance, all the signs are blurred. With the LASIK eye, clarity from the ideal eye is no longer displayed even on the signs that are in focus. As expected, the images produced with the astigmatic eye displays horizontal distortion, but crisp vertical edges.

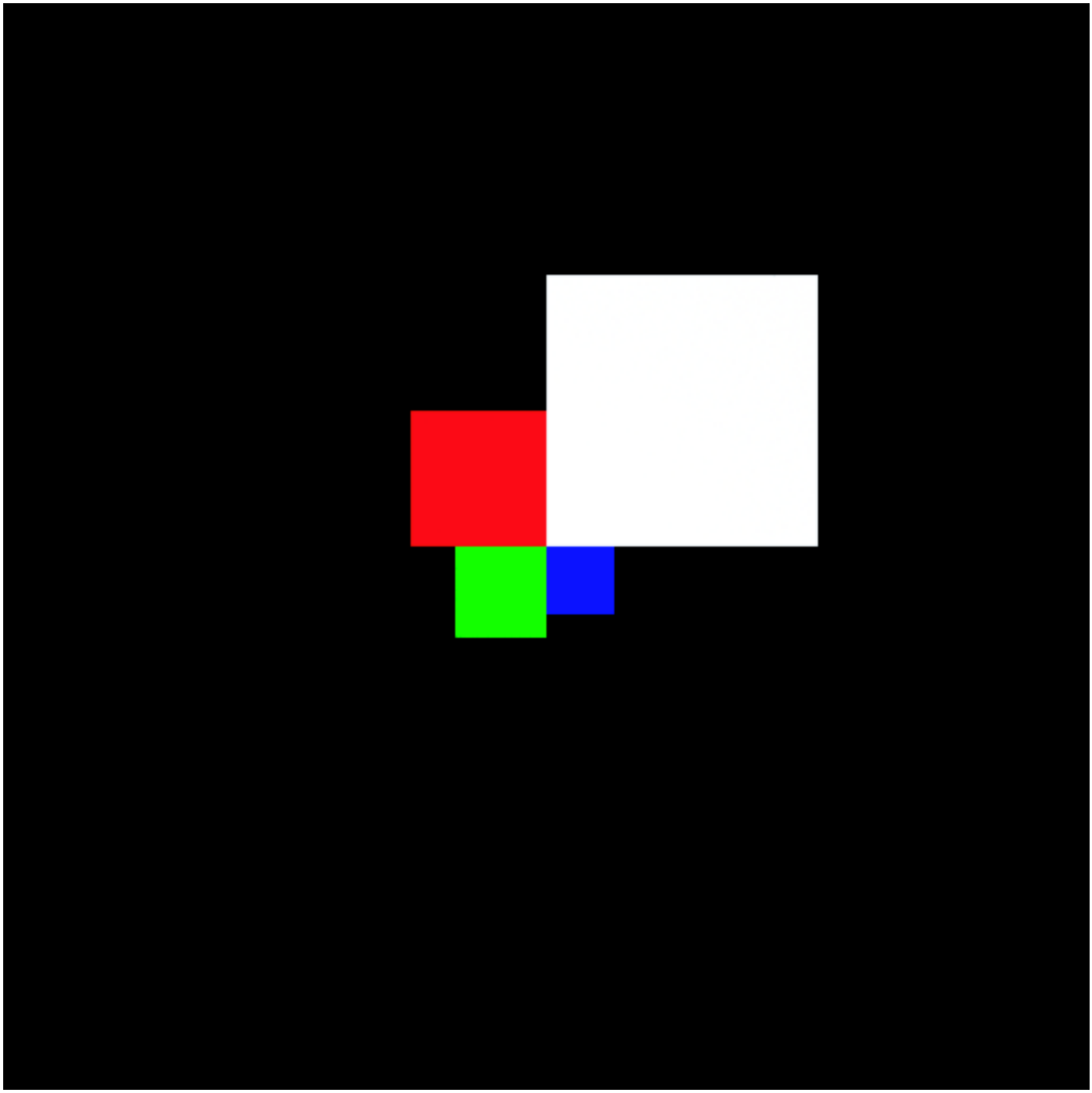
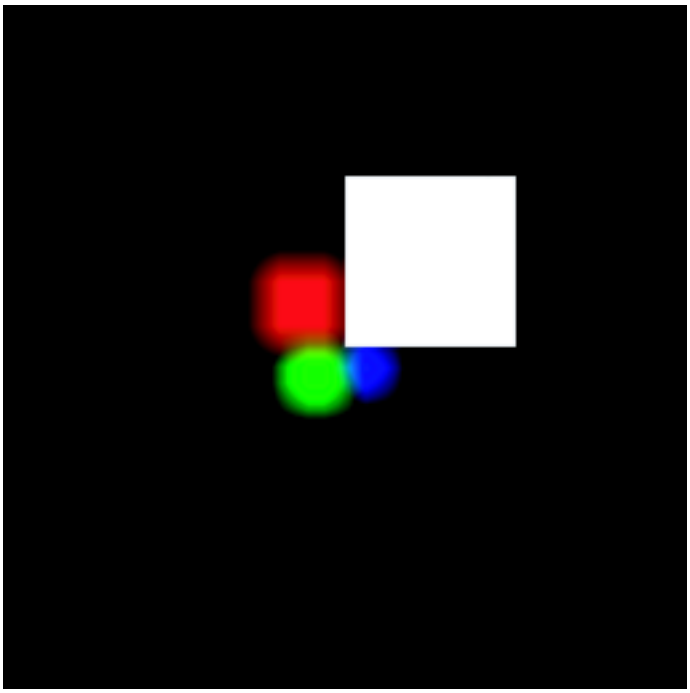
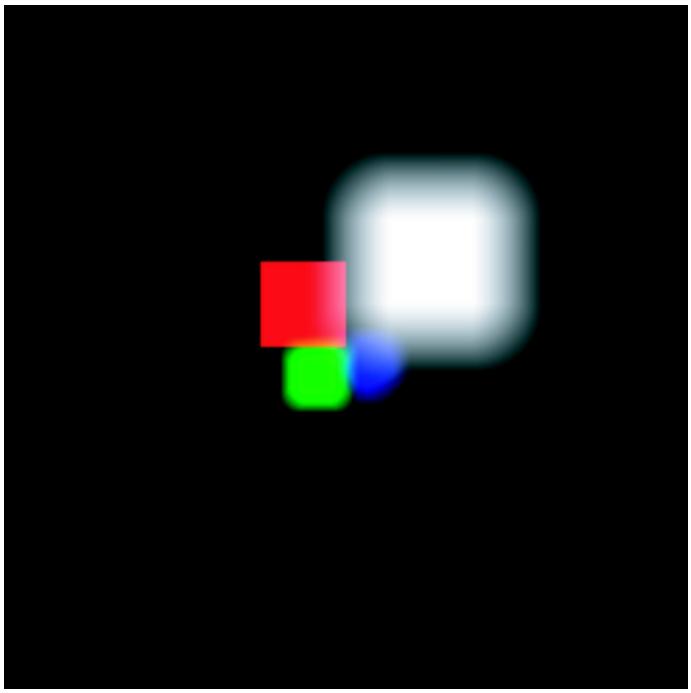


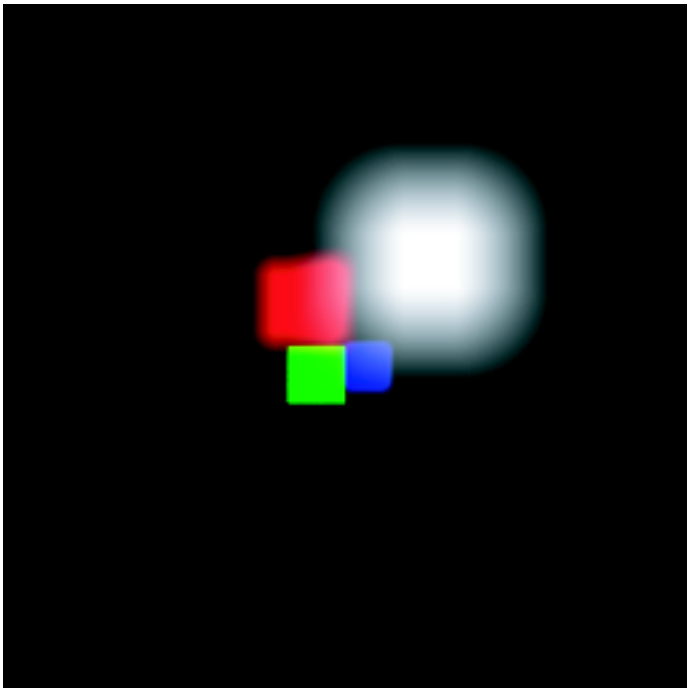
Figure 5.1: Cubes image in its original form.



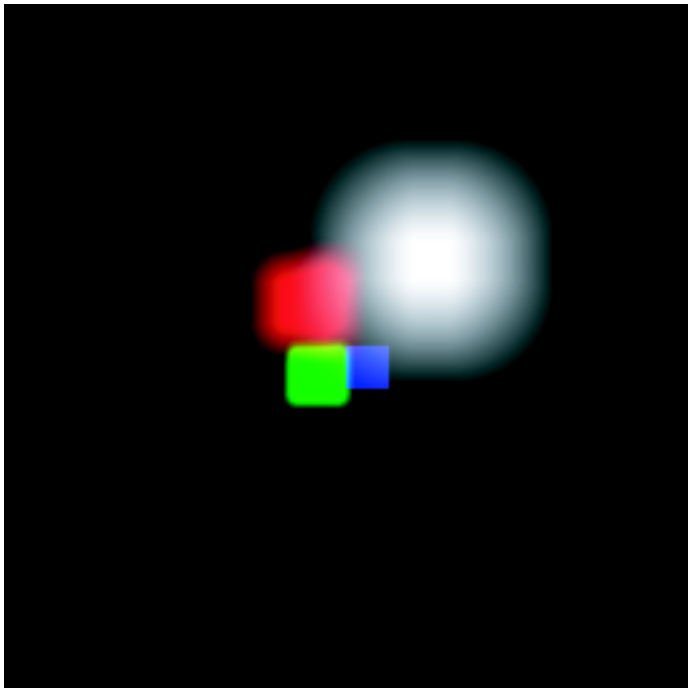
Focused at 0.1 m



Focused at 0.2 m



Focused at 0.3 m



Focused at 0.4 m

Figure 5.2: Cubes image seen through an ideal eye. Each one has a different depth of focus.

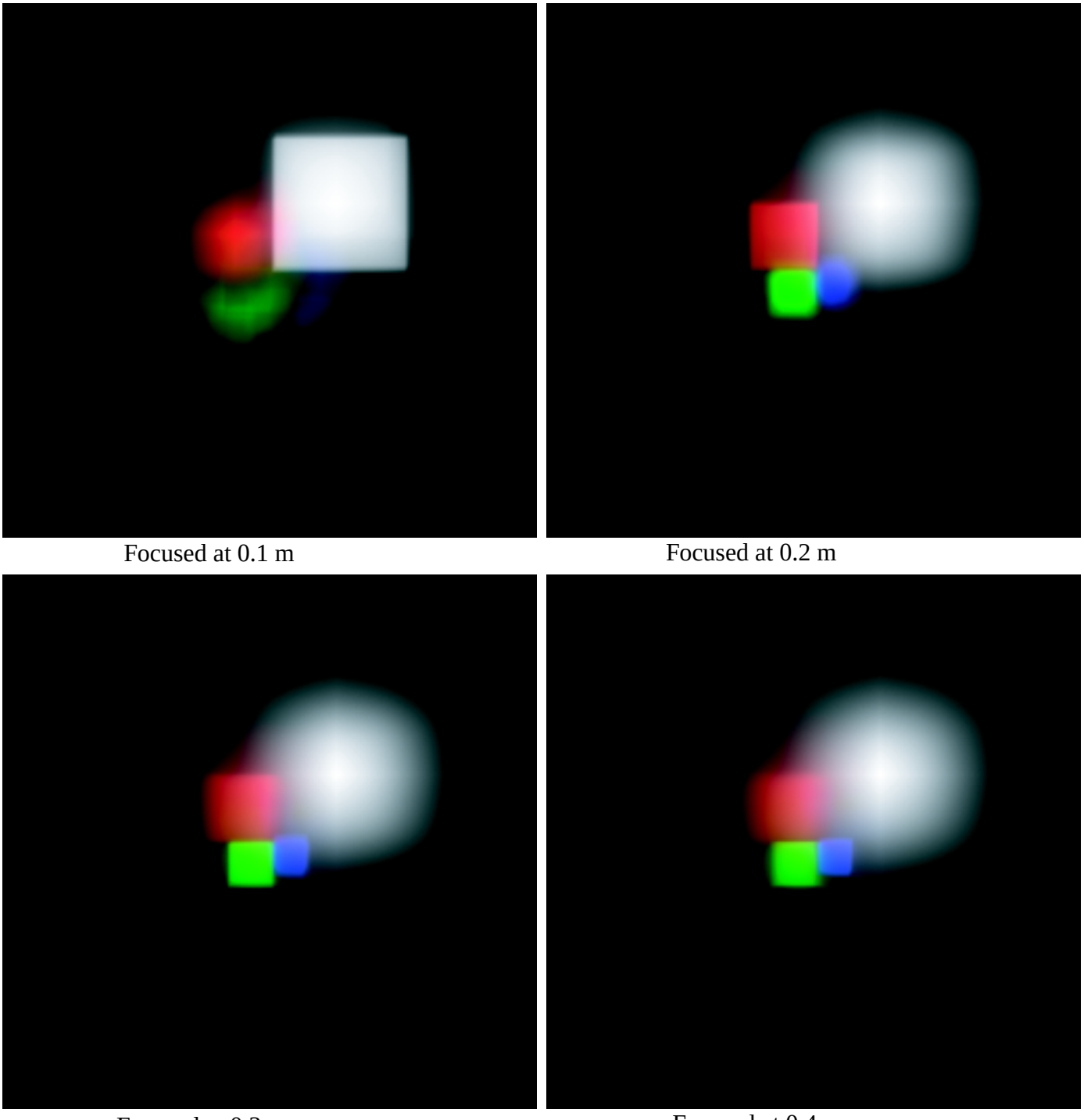
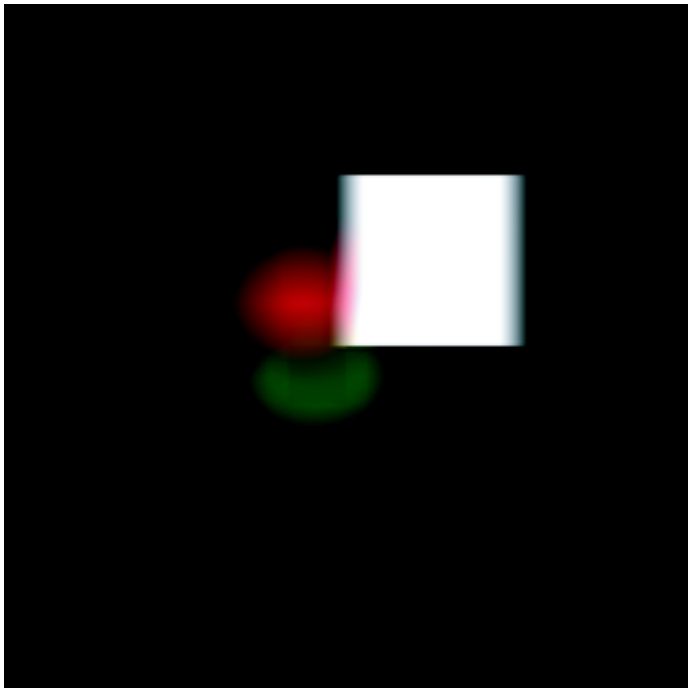
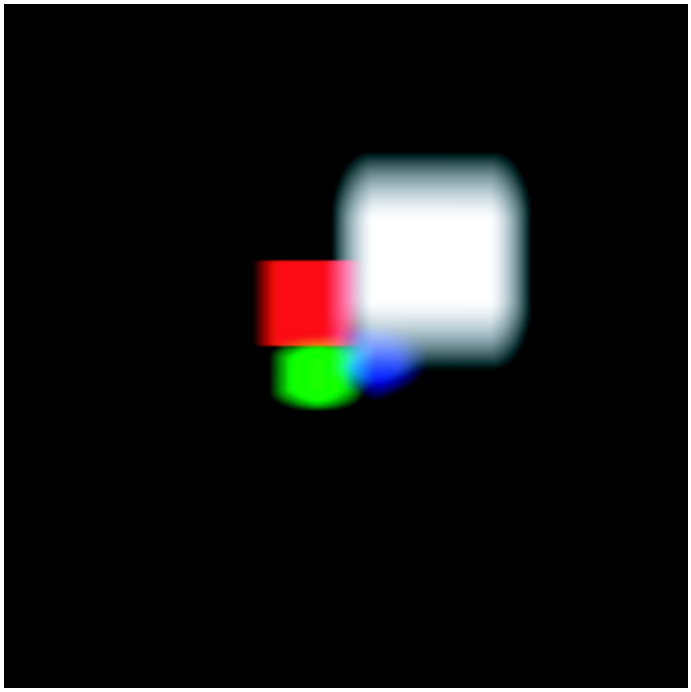


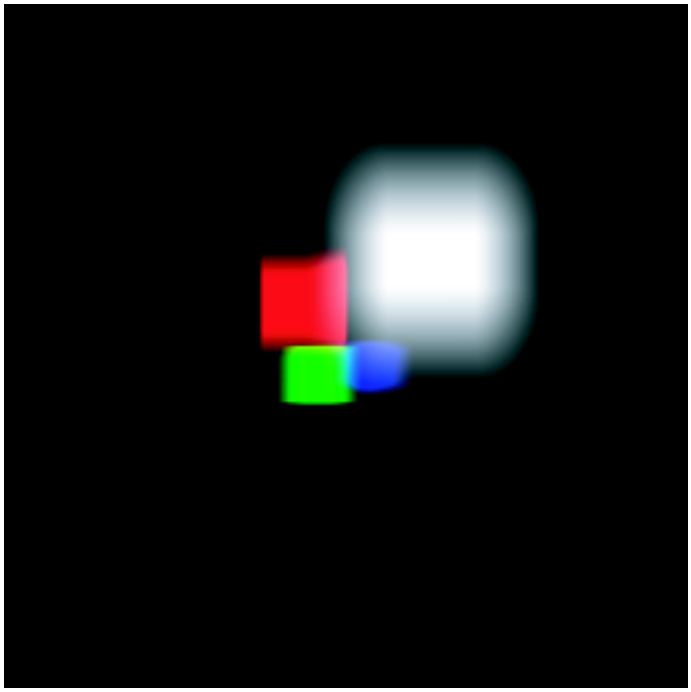
Figure 5.3: Cubes image seen through a LASIK eye. Each one has a different depth of focus.



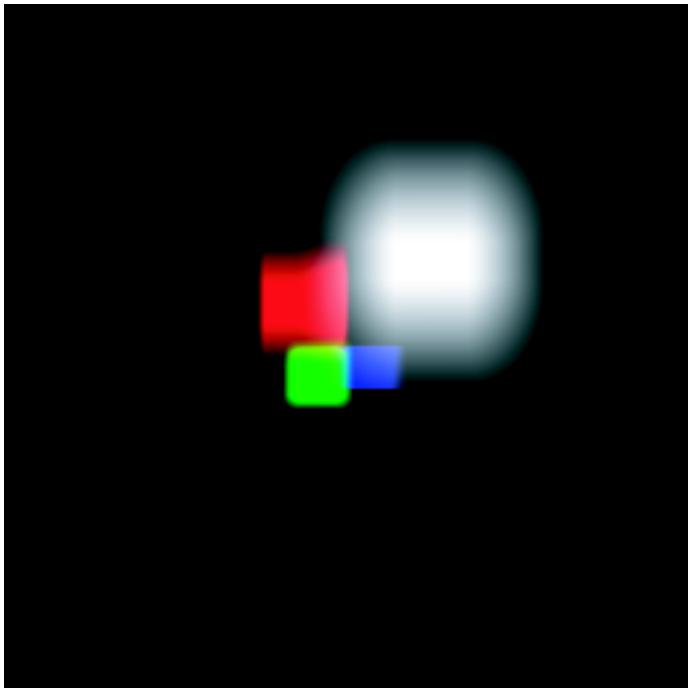
Focused at 0.1 m



Focused at 0.2 m



Focused at 0.3 m



Focused at 0.4 m

Figure 5.4: Cubes image seen through an astigmatic eye. Each one has a different depth of focus.

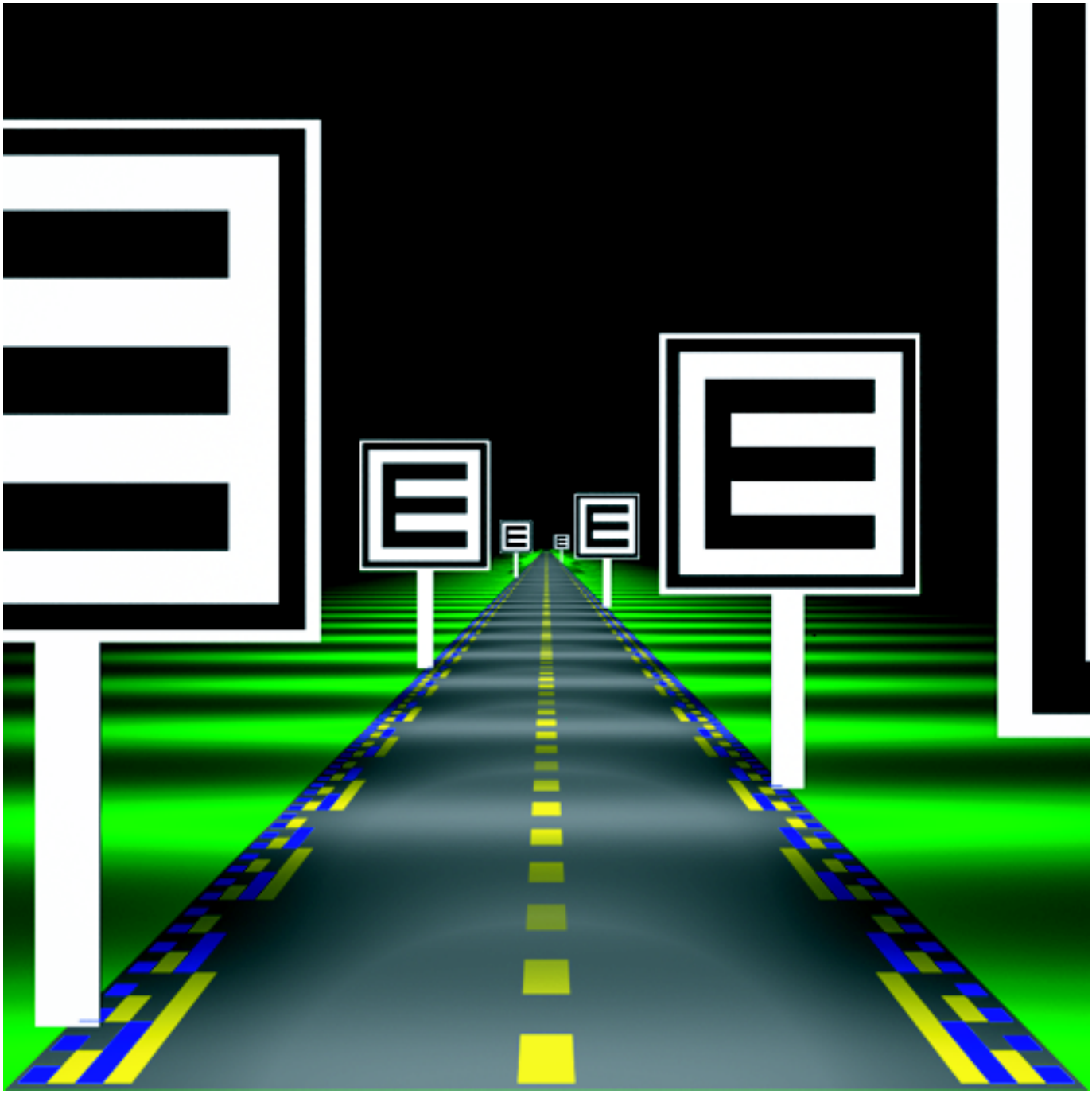
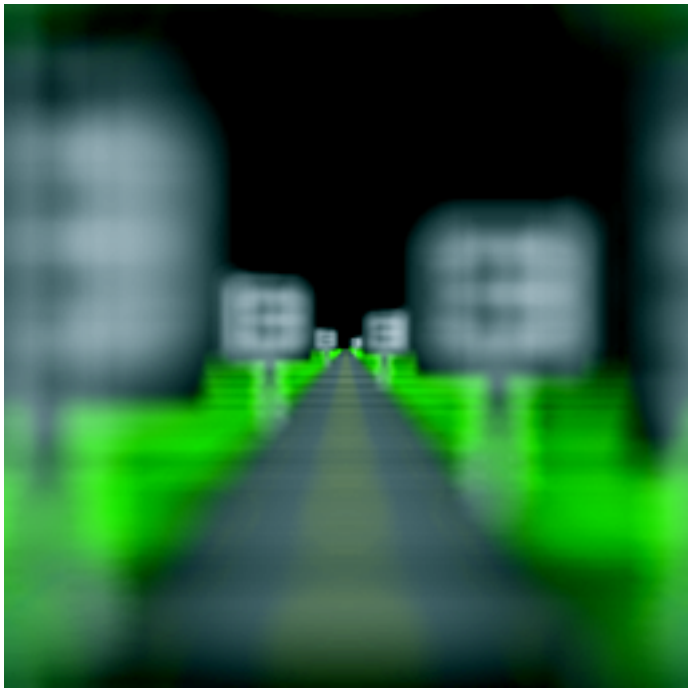


Figure 5.5: Road image in its original form.

Focused at 0.11 m

Focused at 0.22 m

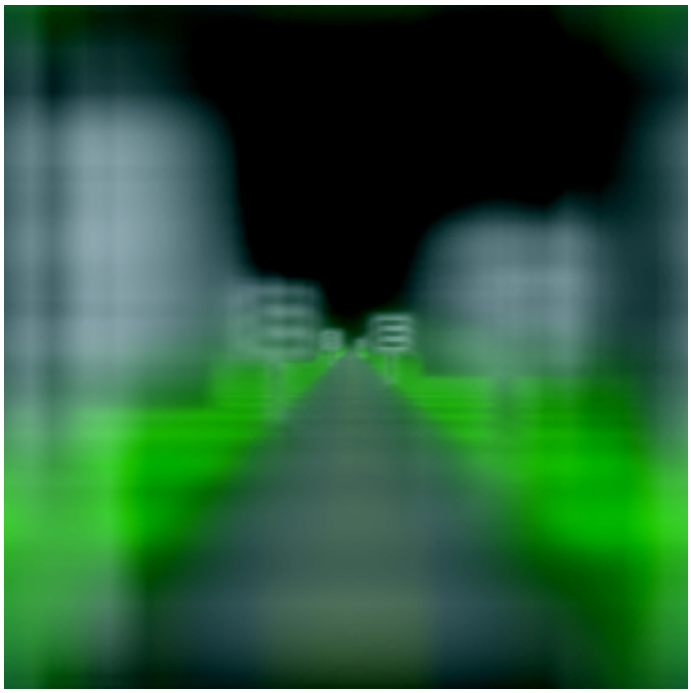


Focused at infinity

Figure 5.6: Road image seen through an ideal eye. Each one has a different depth of focus.

Focused at 0.11 m

Focused at 0.22 m

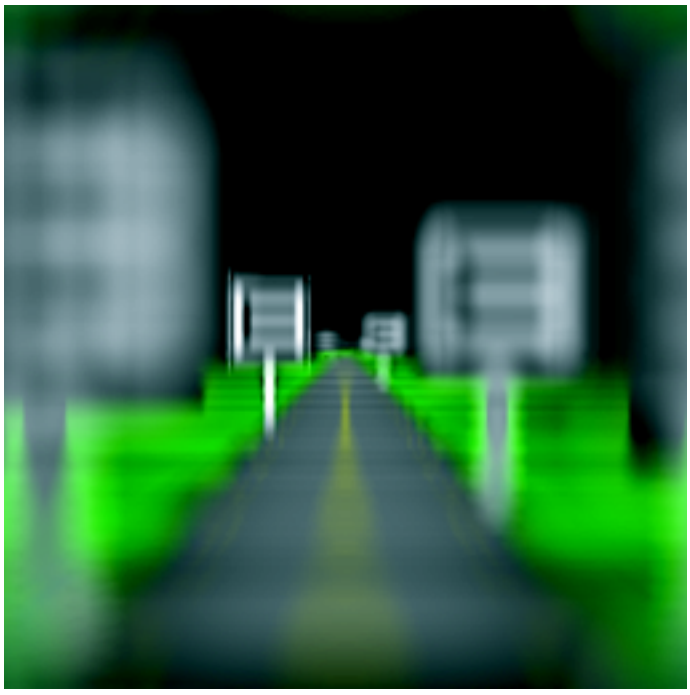


Focused at infinity

Figure 5.7: Road image seen through a LASIK eye. Each one has a different depth of focus.

Focused at 0.11 m

Focused at 0.22 m



Focused at infinity

Figure 5.8: Road image seen through an astigmatic eye. Each one has a different depth of focus.

5.3 Real Life Scene

The image resolution is 400 by 400 pixels, but it was encapsulated in a 512-by-512 image array to take advantage of the FFT algorithm. The FFT method of convolution requires that the dimensions be 2^n where n is a positive number. The image is set in the 8 minutes per pixel world, which means that the scene encompasses ($400*8*\frac{1\ degree}{60\ minutes} = 53.3$) 53.3 by 53.3 degree region in the field of view. The pole on the right side which occludes the background is 0.5 meters away from the viewer. Actual measuring was required to estimate the distances of the objects in the image and they were recorded in the depth file. The differences between images are not as apparent as the scenes rendered with the 0.5 minutes per pixel configuration because using 8 minutes per pixel localizes the histogram at the center of the calculated PSF's, leading to more crisp images.

5.4 Car Scenes

Two car scenes were acquired from an animation along with their depth values. The image resolution is 320 by 240, but is encapsulated in an image array of 512 by 512 pixels to take advantage of the FFT algorithm. The images are set in the 8 minutes per pixel environment. In both scenes, the sign on the left side is more than 8 meters away which, in our algorithm, is the same as being located at an infinite distance. In the first scene, the car in front is 0.5 meters away from the viewer and in the second scene, the car is located 3.85 meters away. With the ideal eye, the distinction is clear between the images focused at the car and at the sign, however, it is not as distinct when the images are viewed with the LASIK eye. Overall blurriness is introduced as expected because of the randomness in

Figure 5.9: Real Life image in its original form.

Focused at 0.5 m

Focused at infinity

Figure 5.10: Real Life image seen through an ideal eye. Each one has a different depth of focus.

Focused at 0.5 m

Focused at infinity

Figure 5.11: Real Life image seen through a LASIK eye. Each one has a different depth of focus.

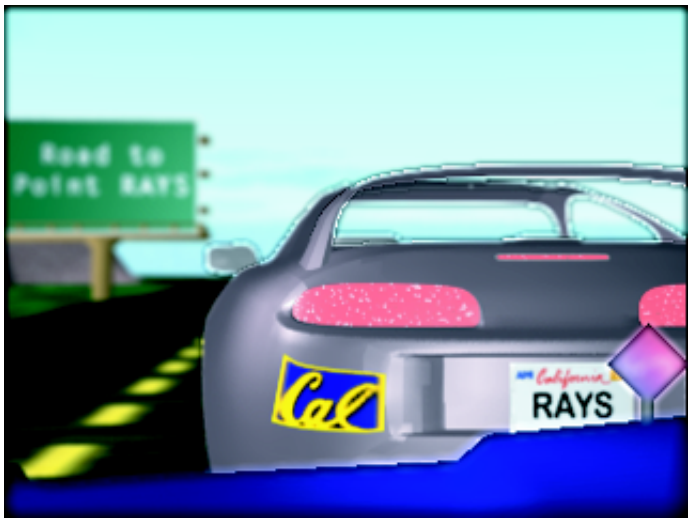
Focused at 0.5 m

Focused at infinity

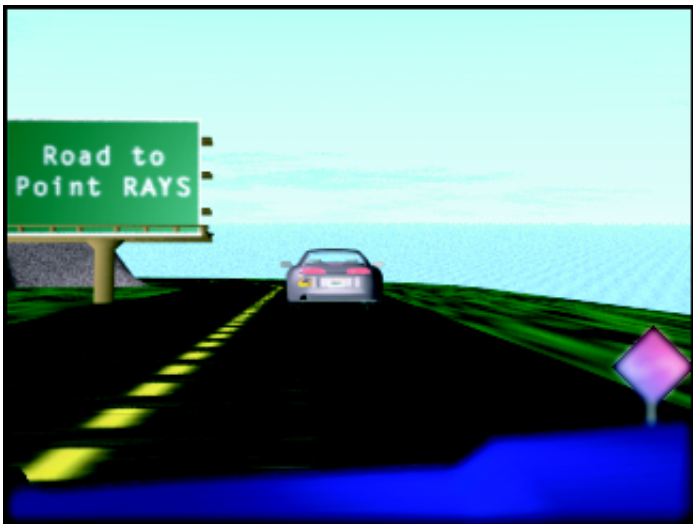
Figure 5.12: Real Life image seen through an astigmatic eye. Each one has a different depth of focus.

the LASIK PSF's. The image produced by the astigmatic eye looks much like the image created by the LASIK eye, but the horizontal distortion is apparent on objects that are in focus.

Figure 5.13: Car images in their original form.



Focused at 0.5 m

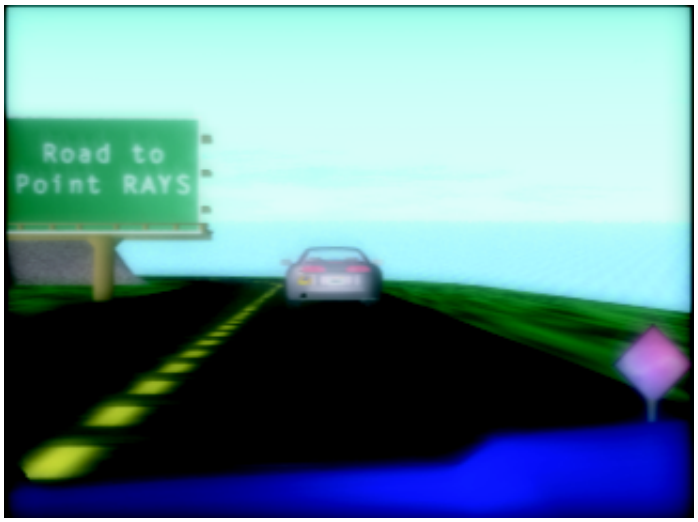


Focused at infinity

Focused at infinity

Figure 5.14: Car images created by the algorithm using the ideal eye model. Each one has a different depth of focus.

Focused at 0.5 m



Focused at infinicty

Focused at infinicty

Figure 5.15: Car images created by the algorithm using the LASIK patient eye model. Each one has a different depth of focus.

Focused at 0.5 m

Focused at infinity

Focused at infinity

Figure 5.16: Car images created by the algorithm using the mathematically induced, astigmatic eye model. Each one has a different depth of focus.

Chapter 6

Conclusion

6.1 Relative Merits of Two Schemes

There are two approaches for approximating the visual acuity of the human visual system. The first method is to use a ray tracer to emit millions of light rays and trace them to calculate every pixel value on the image. The alternative method is what has been described in this thesis: calculating the PSF's that cause precise blurs in the original image and applying them to produce a perception through the visual system.

The ray tracing method has the potential to produce a more realistic image because it computes the resultant pixel value caused by every photon. Currently, this method is implemented with a lens simulating only the corneal surface and using it to provide a comprehensive model of the specified visual system is not possible. The level of accuracy provided by the Shack-Hartmann device in measuring the photons' behavior simply cannot be matched with that of a lens representing just the corneal surface. This method also presents enormous hardware and time requirements because light rays in all possible di-

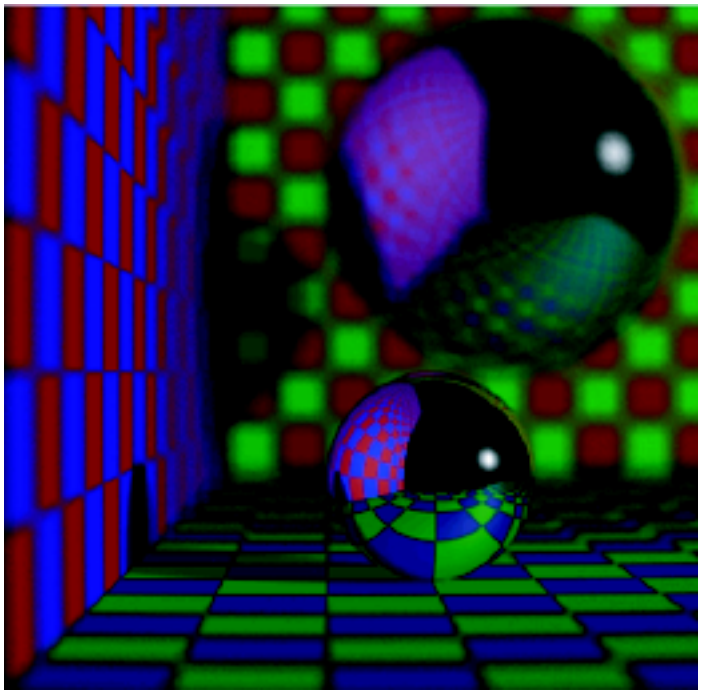


Figure 6.1: A scene rendered using the ray tracing method. The focus is at the bottom metallic ball

rections need to be calculated. Figure 6.1 is an example of a scene created with the ray tracing method.

The PSF method lacks the potential to have the kind of precision presented with the ray tracing method because a set of filters (PSF) is used to estimate the final image values. However, the Shack-Hartmann's measurements are used to implement this approach and while the calculation may not be as precise, the final image is produced with a correct model providing the comprehensive behavior of the visual system. Another advantage of this method is that it requires far less time and system resource.

As displayed in Equation 6.1 and Figure 6.2, the ray tracing method can be mathematically described as integration over the lens representing a corneal surface multiplied by the object space pixel values. The pixel values depend on the x and y coordinates

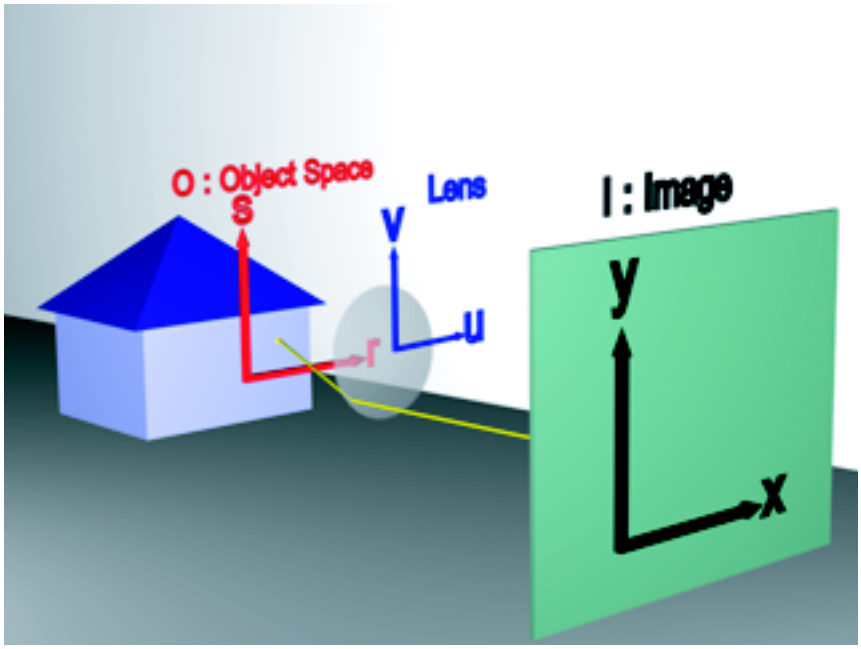


Figure 6.2: Ray tracing method in 3D. The variables in equations 6.1, 6.2, and 6.3 are depicted in this figure.

of object space, which are represented with the variables s and r respectively. To prove the equivalence between the two methods, the equation for the PSF method is presented in Equation 6.3 and substituting $\frac{du}{ds}\frac{dv}{dr}$ in Equation 6.2 with PSF_k gives us the equation for the PSF method where the subscript k is the depth coefficient. These equations are valid under the assumption that the original image is convolved with one PSF, or that there is one depth coefficient for the entire original image. This substitution is possible because the change in u and v respect to s and r ($\frac{du}{ds}\frac{dv}{dr}$) is equivalent to the point spread function.

$$I(x) = \int\int dudv \cdot O(s,r) \tag{6.1}$$

$$= \int\int ds\frac{du}{ds} \cdot dr\frac{dv}{dr} \cdot O(s,r) \tag{6.2}$$

$$I(x) = \int\int dsdr \cdot PSF_k(x-s, y-r) \cdot O_k(s) \tag{6.3}$$

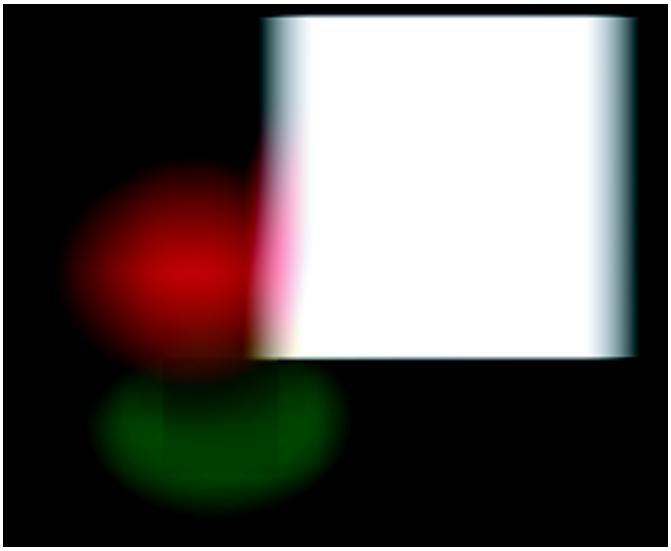


Figure 6.3: The blur from the red cube extends into the white cube.

6.2 Future Directions

Presently, occlusion is not handled properly. Given an image with two objects and the focus at the closer one, distortion from the farther object should not extend into the first object. With our algorithm, the distortion intrudes upon the closer object because the algorithm does not take the depth information into account when the accumulation occurs. This issue could be resolved by implementing a depth ordering unit that recognizes the depths of layers in the stage of accumulation. In Figure 6.3, the blur created by the red cube should not appear on the white cube because it is behind the white cube, but using our algorithm, it extends into the white region.

Pixel saturation is not handled properly in the algorithm. For each pixel, contribution from each depth layer is added to attain the final pixel value. Once the sum of the contribution exceeds the pixel's maximum possible value (255 for 8 bit pixels), further

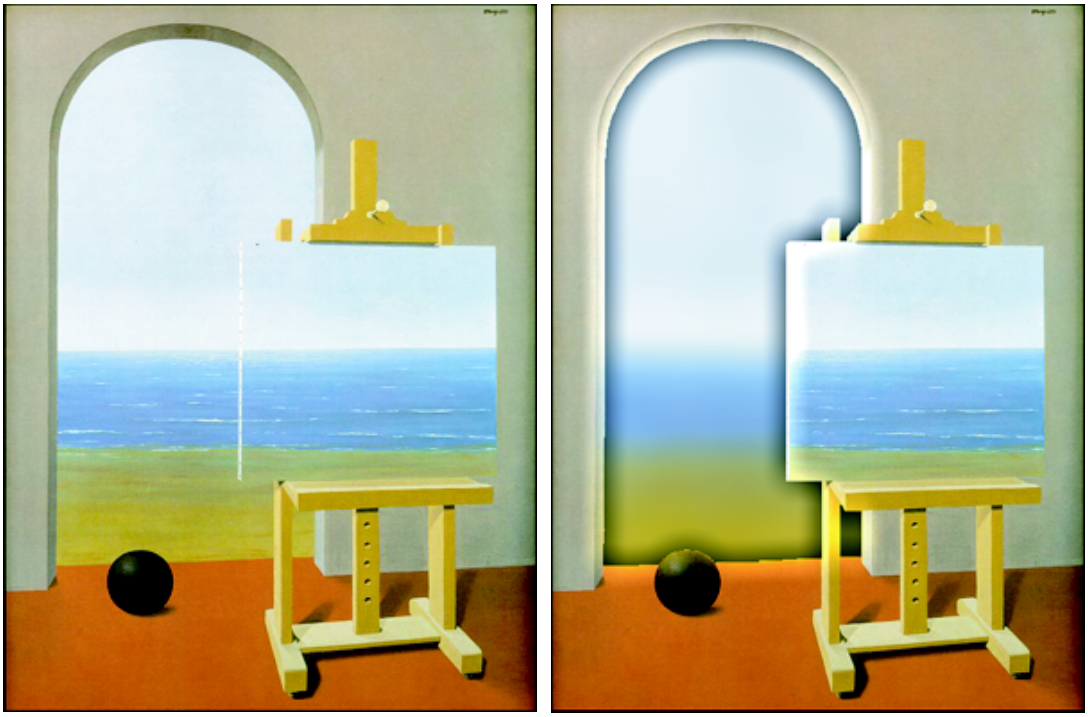


Figure 6.4: The original image on the left. Convolved image on the right and you can see the light and dark bands near the arch.

contribution to the pixel is unnoticed in the final image. This is a common problem when we are handling a scene with an extremely bright light source. This issue can be resolved by using high dynamic range (HDR) or by normalizing the pixel values after accumulation.

In some final scenes, discontinuity between pixels of different depths are observed. What may seem like a discontinuity is actually caused by the dark and light bands near the borders of separated depth images. Figure 6.4 shows an image, seen through an ideal eye whose focus of depth is at the wall, located 0.5 m away. The PSF for the infinite distance contributes energy to the layer of the wall and it results in the light band seen on the outside boundary of the arch. The dark band is created by the loss of "expected energy." The image layer of infinite distance expects contribution from the front layer

Figure 6.5: The original image on the left. Convolved image on the right and the discontinuities can be seen on the road.

that corresponds to 0.5 m, but its PSF is a single peak that does not contribute energy to its neighboring pixels. Implementing the accumulation phase with the alpha channel combination method introduced in '84 SIGGRAPH paper, the problem with the light band might be resolved. The dark band problem might be resolved by preserving the energy for the given depth, similar to what Adobe Photoshop's preserve transparency option.

Figure 6.5 clearly displays the discontinuity problem. The depth layers are distinguishable from one another. A temporary patch was developed for the problem: by convolving the whole image with each PSF and producing the final image by pasting the pixels that correspond to the depth. The current algorithm first isolates pixels for a given depth and then performs convolution. When this isolation occurs, the surrounding pixels have zeros for the RGB values and convolving with zero pixels introduces the dark area

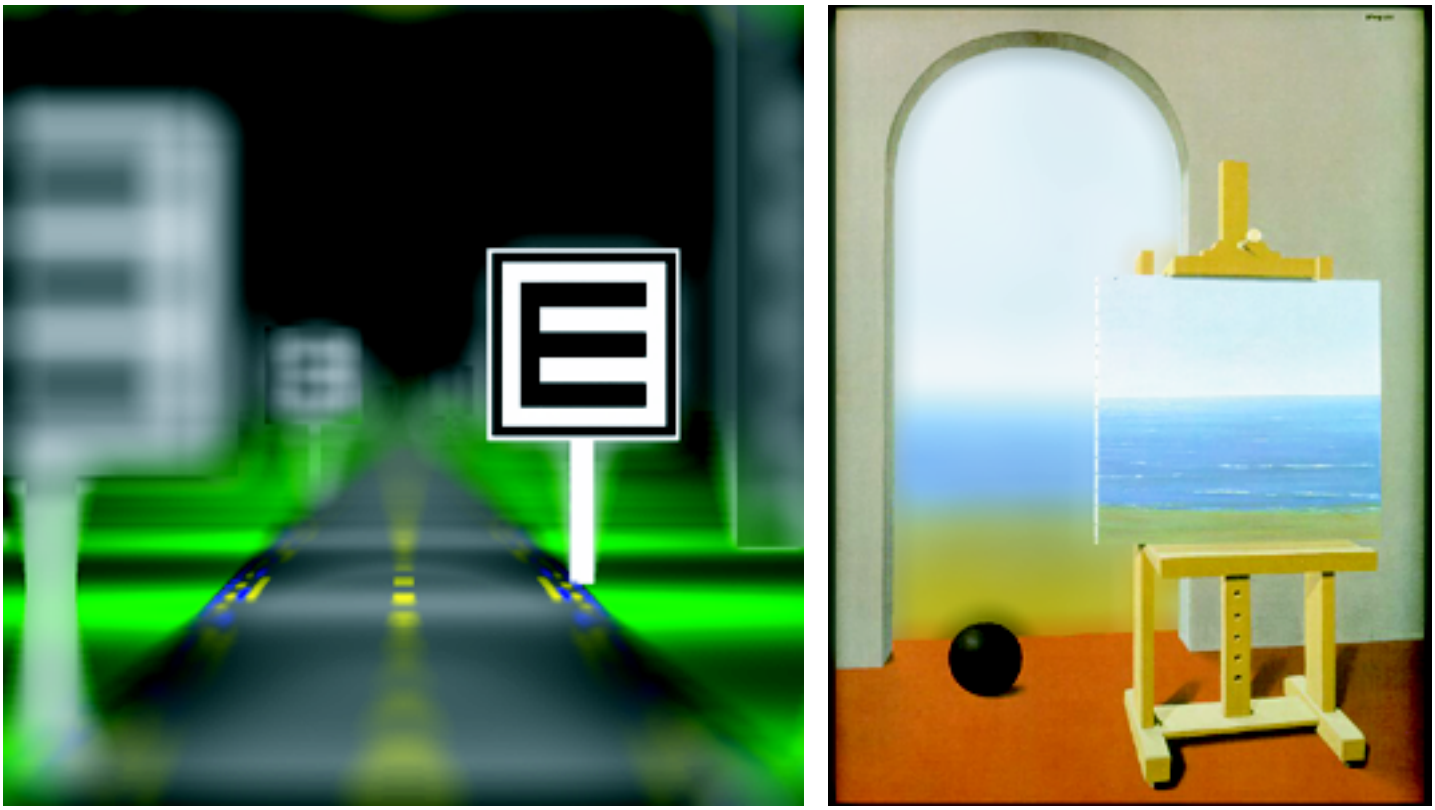


Figure 6.6: Road Scene and the ocean scene convolved with the new algorithm. The striation problem disappears, but some faulty color blending still occurs. For example, the dark area around the ball and the white patches around the signs should not be there.

as can be seen in Figure 6.4 and Figure 6.5. The new algorithm for the patch that we developed performs convolution first, and then isolates the pixels corresponding to each depth. Figure 6.6 contains images produced with the new algorithm. These images do not have the striation problem anymore, but they still have some artifacts that we need to eliminate. In the image on the right, a dark area, which should not be there, is noticed around the ball. It appears in the image because the algorithm does not have the proper 3D information and assumes that the dark ball is at the depth of the ocean.

The discontinuity can also be explained by means of geometric modelling. Using the eye model makes it necessary to gather more information about the input image than what is provided by the pinhole camera model. A complete set of pixel values of each depth is needed for convolution. For example, when convolving an image containing a red balloon in front of the blue sky, a special rendering technique is needed to provide values for the segment of sky behind the balloon. This information is needed to accommodate for the pupil having an actual physical size. The segments that may seem occluded by an object in a pinhole camera model is not considered occluded in the human eye model because those segments contribute to the final image.

Chapter 7

Appendix

7.1 Programmer's Reference Manual

The algorithm is implemented in MATLAB language. These files are laid out in order of execution.

7.1.1 WaveFront.m

```
coef_struct = wavefront(n_factor, file_name)
```

This function calculates the coefficients for Taylor Series expansion which are used to represent the best-fit surface to include Shack-Hartmann data.

- *n_factor*: integer indicating the degree of polynomials used in calculating the best-fit surface. It dictates the number of coefficients. n_factor of 7 was used to acquire the PSF's shown in the Results Section. 7 Produces 37 coefficients.
- *file_name*: name of the text file that contains samples from the Shack-Hartmann device. Each line of the file contains four values: x and y coordinates of the reference

points(green dot), and x and y coordinates of the sub-image(red dot).

- **coef_struct**: output structure containing a coefficient matrix and the n_factor value used to derive the coefficients.

  Fields:

  - **coefs** : The matrix containing the coefficients. The dimensions are n rows by 1 column, where n is the number of coefficients.
  - **factor** : integer that indicates the degree of polynomials. It equals the variable of n_factor.

7.1.2 WaveGen.m

```
resample=wavegen(coef_struct,pupil,sample_size)
```

This function performs re-sampling. Given the coefficients for the low density data, it calculates slopes for re-sampled points.

- *coefstruct*: The output structure from wavefront.m function.
- *pupil*: radius of pupil in millimeter.
- *sample_size*: integer specifying the number of samples needed for re-sampling. Value of 300,000 was used to produce the PSF's shown in Results section.
- **resample**: output matrix that contains the re-sampled wavefront information. Dimensions are 2 $\times$ n rows by 1 column, where n equals the input parameter of sample_size. The first half of the matrix corresponds to aberration information in the x direction and the latter half to aberration information in the y direction.

7.1.3 Calc_PSF.m

`PSF=calc_PSF(depths,S0,resample,pupil,min_per_pix,szPSF)`

This function calculates a set of PSF's that represent a visual system focused at a specified distance.

- *depths*: integer specifying the number of depth planes for the set of PSF structure. Value of 41 was used for images shown in Results Chapter.
- *S0*: diopter measurement of the depth of focus. For example, if the visual system is focused at 0.5 m from the viewer, the value of 2 would be used as a parameter.
- *resample*: The output matrix from wavegen.m function.
- *pupil*: radius of pupil in millimeter. It should equal the value of pupil used in wavegen.m function.
- *min_per_pix*: value specifying the field of view. It specifies the allocation of minutes for each pixel. Values of 0.5 was used to produce the Road and Cubes images, and 8 was used for the Car and Real Life image.
- *szPSF*: integer specifying the dimensions of the collection array. It will produce PSF's whose dimensions are szPSF by szPSF. Value of 127 is used for 8 minutes per pixel images and 255 is used for 0.5 minute per pixel images. This value should be an odd number to preserve the center.
- **PSF**: Output of this function. Three dimensional matrix containing collection of PSF's. The dimensions are n by n by m, where n equals szPSF and m equals the depths parameter in this function.

7.1.4 Make_Image.m

```
output=make_image(f1, f2, PSF)
```

This function performs FFT convolution on a crisp image to produce a blurred image.

- *f1*: original image in TIFF format.
- *f2*: depth file whose dimensions match the original image's.
- *PSF*: The output from calc_PSF.
- **finalimage**: blurred image in TIFF format. The dimensions also match the original image's.

Bibliography

[1] Pattern Recognition Group at Delft University of Technology. Convolution-based operations. http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip-Convolut-2.html#Heading85.

[2] Brian A. Barsky, Billy P. Chen, Alexander C. Berg, Maxene Moutet, Daniel D. Garcia, and Stanley Klein. Incorporating Camera Models, Ocular Models, and Actual Patient Eye Data for Photo-Realistic and Vision-Realistic Rendering. *Mathematical Methods in CAGD: Oslo 2000, Oslo, Norway*, 2000.

[3] Brian A. Barsky, Daniel D. Garcia, Stanley A. Klein, Woojin M. Yu, Billy P. Chen, and Sarang S. Dalal. RAYS (Render As You See): Vision-Realistic Rendering Using Hartmann-Shack Wavefront Aberrations. *Internal Report*, 2001.

[4] A. M. Bas and R. Onnis. Excimer LASIK for Myopia. *Journal of Refractive Surgery*, 11(suppl):229–233, 1995.

[5] Lucio Buratto and Stephen Brint. *LASIK Surgical Techniques and Complications*. SLACK Incorporated, Thorofare, NJ, 2000.

[6] C.S. Burrus and T.W. Parks. *DFT/FFT and Convolution Algorithms Theory and Implementation.* John Wiley & Sons, Houston, Texas, 1985.

[7] Silicon Valley Medical Center. Vision correction. http://www.perfecteyesight.com/.

[8] Daniel D. Garcia. *CWhatUC : Software Tools for Predicting, Visualizing and Simulating Corneal Visual Acuity.* Ph.D. Thesis, University of California, Berkeley, CA, May 2000.

[9] David Halliday and Robert Resnick. *Physics, Part Two.* John Wiley and Sons, Inc., New York, NY, 1978.

[10] National Eye Institute. The cornea and corneal disease. http://www.nei.nih.gov/publications/cornea.htm.

[11] Craig Kolb, Don Mitchell, and Pat Hanrahan. A Realistic Camera Model for Computer Graphics. In *Proceedings of ACM SIGGRAPH 1995*, pages 317–324, Los Angeles, CA, August 6-11 1995. SIGGRAPH, Association of Computer Machinery, Inc.

[12] Jay H. Krachmer, Mark J. Mannis, and Ednad J. Holland. *Cornea, Surgery of the Cornea and Conjunctiva*, volume 3. Mosby-Year Book, Inc., St. Louis, MO, 1997.

[13] Junzhong Liang. *A New Method to Precisely Measure the Wave Aberrations of the Human Eye with a Hartmann-Shack Wavefront Sensor.* Ph.D. Thesis, Department of Mathematics, Universität Heidelberg, Heidelberg, Germany, December 1991.

[14] Jae S. Lim. *Two-Dimensional Signal and Image Processing.* Prentice Hall, Englewood Cliffs, NJ, 1990.

[15] Microsoft. Encarta. CD-ROM, 1999. Multimedia Encyclopedia under section Eye.

[16] John Morgan. LASIK surgery gives Tiger Eagle Eyes. *USA Today*, 3/15/2000, 2000.

[17] Clyde W. Oyster. *The Human Eye Structure and Function.* Sinquer Associates Publishers, Inc., Sunderland, MA, 1999.

[18] I.G. Pallikaris, M.E. Papatzanaki, and E.Z. Stathi. Laser In Situ Keratomileusis. *Lasers Surg. Med.*, 10(3):463–468, 1990.

[19] I.G. Pallikaris and D.S. Siganos. Excimer Laser In Situ Keratomileusis And Photorefractive Keratectomy. *Journal of Refractive Surgery*, 10:498–510, 1994.

[20] Juan J. Perez-Santonja, Juan Bellot, Pascual Claramonte, Mahmond M. Ismail, and Jorge L. Allo. Laser In Situ Keratomileusis to Correct High Myopia. *Journal of Cataract and Refractive Surgery*, 23:372–385, 1997 (Jan-Jun).

[21] Ben Platt and Roland V. Shack. Lenticular Hartmann-screen. Newsletter 5, 15, Optical Science Center, University of Arizona, 1971.

[22] Wave Front Sciences. The shack-hartmann sensor. http://www.wavefrontsciences.com/ophthalmic/hartmannsensor.html.

[23] Larry N. Thibos. Principles of Hartmann-Shack Aberrometry. In V. Lakshminarayanan, editor, *Vision Science and its Applications*, volume 15, pages 163–169. OSA-TOPS, Washington, DC, 2000.