NUMERICAL TECHNIQUES FOR
FRESNEL DIFFRACTION IN
COMPUTATIONAL HOLOGRAPHY

A Dissertation

Submitted to the Graduate Faculty of
the Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

in

The Department of Computer Science

by
Richard Patrick Muffoletto
B.S., Physics, Louisiana State University, 2001
B.S., Computer Science, Louisiana State University, 2001
December 2006

Acknowledgments

Table of Contents

List of Tables

List of Figures

Abstract

Optical holography can produce very realistic virtual images due to its capability to properly convey the depth cues that we use to interpret three-dimensional objects. Computational holography is the use of digital representations plus computational methods to carry out the holographic operations of construction and reconstruction. The large computational requirements of holographic simulations prohibit present-day existence of real-time holographic displays comparable in size to traditional two-dimensional displays.

Fourier-based approaches to calculate the Fresnel diffraction of light provide one of the most efficient algorithms for holographic computations because this permits the use of the fast Fourier transform (FFT). The limitations on sampling imposed by Fourier-based algorithms have been overcome by the development, in this research, of a fast shifted Fresnel transform. This fast shifted Fresnel transform was used to develop a tiling approach to hologram construction and reconstruction, which computes the Fresnel propagation of light between parallel planes having different resolutions. A new method for hologram construction is presented, named partitioned hologram computation, which applies the concepts of the shifted Fresnel transform and tiling.

Chapter 1

Introduction

Optical holography is a very realistic form of visualization. Providing all of the depth cues that the human visual system uses for comprehending three dimensional objects, holograms can produce virtual images that are indistinguishable from real objects [Har96, Fow89]. The strongest visual cue is binocular disparity, which is the result of different perspectives from a three-dimensional scene reaching each eye. Our brain interprets these two perspectives and forms an understanding of the three-dimensional scene. Traditional two-dimensional imaging is capable of conveying many of the depth cues associated with three dimensional space, yet lacks the ability to produce binocular disparity.

Traditional optical holography involves the recording of reflected laser light onto special photographic plates [Fow89]. Computational holography is the use of digital representations and computational methods to carry out holographic operations; modeling the propagation of complex-valued wavefronts through three-dimensional space [LCR02]. The two major steps in holography are the creation of the hologram (the fringe pattern) and the reconstruction (or display) of the hologram. Either of these steps can be carried out computationally, yet the complexity involved is daunting. The amount of information stored in a hologram and the computational effort that goes into its calculation are several orders of magnitude beyond traditional two-dimensional imaging [SHBL92].

Various approaches to computational hologram construction have been researched, of which the most prominant are interference-based [MT00, IS04, HMHG05], Fourier-based [NOY$^+$97, AR03, GMD96], and diffraction-specific [Luc94, Luc96] hologram computation. This research addresses the Fourier-based approach. Fourier-based hologram computation is built around the fast Fourier transform (FFT), which provides efficient computational algorithms. The disadvantages of traditional Fourier techniques are that the sampling rates are fixed and that the sampled points of the hologram and object must both be centered along the axis of propagation [MGF$^+$99, OO05]. One main contribution of this research is the development of fast computational methods to construct and reconstruct holograms with arbitrary sampling rates and with a spatial shift away from the axis of propagation.

This dissertation introduces the concepts of holography in Chapter 2 and covers the mathematical modeling involved in computational holography in Chapter 3. Chapter 4 introduces and derives a new form of the Fresnel diffraction integral that solves a problem with traditional Fourier-based hologram computation by allowing for an arbitrary spatial shift of the wavefront. Two applications of shifted Fresnel diffraction are presented and explored. Chapter 5 introduces the concept of tiling, which allows for propagation of discrete wavefronts between sampled planes having different resolutions. Chapter 6 presents a new technique for hologram construction, referred to as partitioned hologram computation, which applies the concepts of spatially shifted Fresnel diffraction and tiling. An overview of our results and possible future enhancements to this research are presented in Chapter 7.

Chapter 2

Holography Concepts

2.1   Holography: The Ultimate Visualization Tool

2.1.1   Seeing in 3D

Most would agree that holography produces the most realistic form of visualization. The striking realism of a well crafted hologram can be breathtaking. The same physical wavefront of light that we see from a real object can be reproduced by a hologram. Thus viewing a hologram is no different than viewing the actual object which it represents. In the comprehension of a three-dimensional (3D) scene, the human visual system relies on many depth cues, all of which can be reproduced by holography. To better understand the mechanisms behind this realism, the following presents some specific methods through which one may comprehend 3D space. For a more in-depth discussion of these topics, the interested reader may see [Luc94, How95, DD90, Boo01].

The Monocular cues can all be seen with one eye, and can be produced with simple two-dimensional (2D) imaging. Occlusion is the layering effect whereby objects in the foreground overlap objects in the background. Pictorial depth cues are a result of perspective rendering and proper shading. The most dynamic monocular cue is motion parallax; the principle through which objects that are further from you appear to have less lateral movement than closer objects. This can be accomplished in 2D by the simple rotation or translation of a scene, and by properly including the parallax effect into the rendering process.

Binocular disparity, or stereopsis, occurs when each eye views a scene from a slightly different angle. Our brain is highly efficient at meshing two 2D scenes into one coherent depth-perceived visual response. This is perhaps the most striking depth cue, and is why holograms seem so real. There have been techniques developed, other than holography, that are capable of producing binocular disparity and will be presented in Section 2.1.2.

A small percentage of the population actually relies very little, if at all, on binocular disparity. It is believed that various reasons lead to this condition (either a malfunctioning of one of the eyes or a neurological defect), which is referred to as stereoscopic blindness [Ric70, MSLC05]. Those afflicted rely more on 2D cues, such as texture or shading, and favor one eye much more than the other. For this portion of the population, holography unfortunately has little to offer over simple 2D rendering of 3D information.

2.1.2   Other 3D Visualization Systems

As promising as holographic visualization is, it is not without its share of technical challenges — some of which will be explored in later sections. Because of this, research into other 3D visualization techniques has progressed much quicker and has produced more viable solutions which are readily applicable in real world situations. As previously discussed, simple 2D imaging is capable of conveying many of the depth cues the human visual system associates with 3D vision. Binocular disparity, lacking in a single

2D image, can be conveyed through the use of stereoscopic or autostereoscopic systems. The purpose of these systems is to somehow deliver a different image to each eye. These two images, referred to as a stereo-image pair, would be generated from a 3D scene and correspond to two perspectives having an angular separation roughly equal to that of our eyes. When these images are processed separately by each eye, our brain interprets them as part of a true 3D scene. The difference between stereoscopic and autostereoscopic systems is simply the means through which they accomplish binocular disparity.

A stereoscopic system employs hardware, such as goggles and/or a head-tracking system, with which the user interacts to determine which 2D images to generate [MRCD02, GHTI98, MJvR03]. At any one time, the system only needs to generate two 2D images, thus the computational needs are only twice that of a traditional 2D rendering system. Head-tracking systems, while more complicated, can greatly increase the natural feel of interaction with a 3D environment. The major drawback of this approach is any hindrance on the user caused by the hardware.

An autostereoscopic system also presents a different rendering to each eye, but does not require the viewer to directly interact with hardware such as goggles. Holography is one type of autostereoscopic system. Other, more mechanical, systems typically use some shutter mechanism or a directional display that direct the two 2D renderings to the proper eye [Har96]. One such implementation displays an interleaved stereo-image pair, produced from two different perspectives of a scene, onto an LCD pixeled array that is located behind a special diffractive optical element which directs the proper columns of the digital image toward the proper eye [JNK⁺95]. Another approach is based on the time-multiplexing of various perspectives, each of which is projected using a micromirror array [YKCA01, YKC⁺04]. A common drawback of these systems is that only horizontal parallax is produced — a tradeoff sometimes taken to reduce computational complexity and simplify the system design. A more recent advancement, referred to as integral imaging, uses a diffractive lens array to allow for both horizontal and vertical parallax [CKLL05, HvSS⁺05, LIHD04, MCJMCS05, NYH01, PKK⁺04].

The approaches and techniques presented above either sacrifice depth cues or rely on some type of hardware that severely limits the number of viewers. They are not true holographic systems, but they do represent crucial developments toward such a system. As research and development continue, we expect more sophisticated 3D visualization systems that more closely resemble true holography — eventually blurring any distinction between the two.

2.2 A Brief History of Holography and its Applications

The ideas behind holography were first invented by Dennis Gabor in 1947 for use in wavefront reconstruction in the field of electron microscopy [Gab48]. He was trying to remove aberration issues to improve the resolution of electron microscopes. Gabor coined the term hologram, wishing to convey that it contained all information about the object's wavefront, i.e. both the amplitude and the phase. The concept was later independently discovered by Leith and Upatnieks [LU64] in the United States and Denisyuk in Russia [LCR02].

One of the first highly successful applications of holography was the development of holographic optical elements (HOE), wherein a diffractive fringe pattern could be made to serve as a lens, a grating, or even a beam splitter [LCR02]. HOEs are especially useful in specialized applications because they

can be designed to reproduce any complex-valued wavefront (see [Har96] for a list of some interesting applications).

Another early application was holographic interferometry which began in 1965. It was discovered that holography could be used to investigate the behavior of a moving object. If the object moved during the exposure, e.g. perhaps just a simple vibration, the reconstructed hologram would exhibit speckle patterns showing its movement [LCR02, LBB$^+$05]. More recently, holographic interferometry has been used in noninvasive medical imaging [GAD$^+$02, MRM$^+$05, Pir02]. Light having a short coherence length is passed through biological tissue and made to interfere with a reference beam. The light that passes through the material the fastest, i.e. scattering the least, interferes the best with the reference beam, and therefore is capable of reconstructing the medium through which it passed better.

Holography has also been used to create more efficient solar energy components [Sto93, FWS$^+$92], optical information storage [LHCW00, MDM$^+$05], microscopy [YKOM01], the testing of optical elements [MW71], and encryption systems [TJ00, HS03, WZJW06]. A common thread we see in these applications is that some concept of holography has been built on and expanded, where many times the final technique bears little resemblance to holography. As research continues forward, Leith believes that the "boundary between holography and nonholography will become more indistinct [LCR02]."

In this research, our chief focus is optical display holography, even though some techniques developed could possibly be applied in other applications of holography.

2.3 Optical Holography Principles

There have been many applications of holographic techniques, as discussed in Section 2.2, but not all of them produce what most would think of as a hologram. Optical, or display, holography typically refers to holographic techniques involved in producing three-dimensional images through diffractive principles. In this section, we will briefly explain diffraction, and how optical holograms are created, recorded, and viewed.

2.3.1 Diffraction

Classical geometry can be used to explain how light that is blocked by an opaque structure would cast a shadow with the exact outline of that structure. But this is not what we see. The boundary of a shadow is fuzzy — some of the light crosses into the shadowed region and some of the "darkness" crosses into the lit region. The same principle is observed as light passes through smaller apertures, e.g. Young's double slit experiment [Fow89]. These deviations from what one would expect from classical geometrical optics are lumped into a phenomenon referred to as diffraction. The same properties are observed when water waves pass an obstacle, or sound waves pass through a doorway.

Many times the key to observing diffraction, as with holography, is that the diffracting boundary or aperture has dimensions on the order of the wavelength of the wave in question. Thus for holography, we would expect to need apertures on the order of the wavelength of visible light, i.e. a range from approximately 380nm to 780nm which depends on the reference used [Roh94, CO96, Lap78, Fow89]).

2.3.2 Optical Hologram Construction

A traditional optical hologram of an object is produced using a coherent monochromatic light source, i.e. it produces light of only one wavelength and all light leaving the source at each instant has the same phase. Figure 2.1 shows a typical hologram recording configuration.
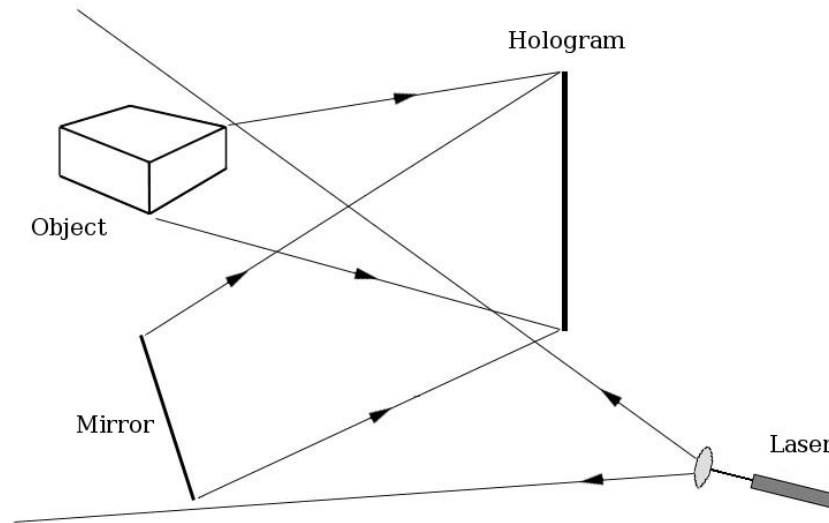
Figure 2.1: Typical optical hologram construction. The light reflected from the mirror interferes with light reflected from the object.

The light source is split into two beams, usually through the use of mirrors or lenses. One beam acts as a reference beam and is directed at a piece of film (photographic plate); the other beam is reflected off an object and then directed to the film. The two beams, or wavefronts, interact at the surface of the film to produce a standing wave interference pattern — also known as a holographic fringe pattern. The standing wave pattern produced is a result of the linear superposition of the light waves. The waves will either constructively or destructively interfere; their interference is based on the phase of the light interacting. Figure 2.2 illustrates the interference of two wave patterns.
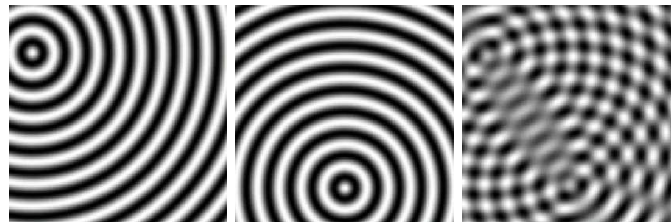
Figure 2.2: Interference of two wave patterns. The images on the left and in the center were summed to produce the image on the right.

Early optical holography, termed on-axis holography, used a reference beam directed perpendicular to the film [Gab48]. The object would be a transparency consisting of opaque structures with a clear background. Interference would occur at the film between the unhindered reference beam, and the light

scattered from the opaque structures in the transparency. The main disadvantage of on-axis holography is that during reconstruction, both the real and virtual images (to be discussed in Section 2.3.3) are centered about the axis normal to the hologram. Thus when viewing the virtual image, an out-of-focus real image is superimposed, and vice versa.

This major shortcoming of on-axis holography was first solved by Leith and Upatnieks [LU62, LU64]. They discovered that inclining the reference beam to the hologram allowed for a reconstruction consisting of angularly separated real and virtual images. This method is referred to as off-axis holography, which was illustrated in Figure 2.1. Since the reference beam is inclined, it has a spatially varying phase across the hologram. Hariharan [Har96] explains this quite well, stating that "the amplitude and phase of the object wave are therefore encoded as amplitude and phase modulation, respectively, of a set of interference fringes equivalent to a carrier with a spatial frequency [equal to that of the reference beam]."

2.3.3 Optical Hologram Reconstruction

One truly impressive aspect of holography is produced in the reconstruction step. The reconstruction of a hologram produces two representations of the original object, termed the real image and the virtual image. In either case, the reconstruction step begins by illuminating the hologram with a reference beam having the same wavelength as the one used to produce the hologram. When the illumination angle is the same as during the construction step, the original wavefront produced by the object is reproduced. The film is acting as a diffraction pattern, bending the light in such a way that it mimics the original wavefront recorded from the object.

Any region of a hologram, however small, theoretically contains all information necessary for reconstruction of the object because that region is dependent on the entire object. The analogy of a hologram to a window is often made. Covering a portion still allows the entire object to be seen, except perhaps from a reduced viewing angle [Har96].

The real image is formed from light being diffracted from the hologram that converges at a viewing plane. The reconstructed image, termed pseudoscopic, is displayed with depth inversion. The reference beam used, termed a conjugate reference beam, is directed at the opposite angle of the original beam. During real image reconstruction, only a small portion of the hologram is illuminated. Otherwise many perspectives of the object would be produced at the same time, each converging on the viewing plane, and thus reducing the clarity of the reconstruction. A typical setup for real image hologram reconstruction is illustrated in Figure 2.3.

The virtual image is produced by the light being diffracted from the hologram that appears to diverge from the location of the original object (see Figure 2.4). The reconstruction produces all of the original depth cues of the object. The reference beam is spread out in such a way as to illuminate the entire hologram. Thus, every perspective recorded in the hologram is reconstructed at the same instant, and the user is free to view the hologram from any angle. The diffracted light again constructively or destructively interferes upon reaching the eye, but the eye interprets this light as diverging from three-dimensional locations representative of the original object. Since each eye is looking into a different region of the hologram, it sees a different view of the object which results in binocular disparity [Har96].
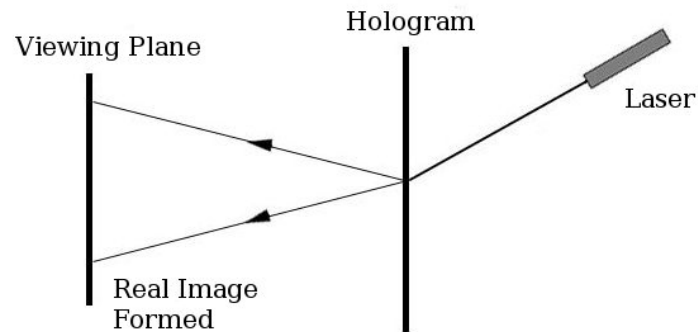
Figure 2.3: Typical optical hologram real image reconstruction. The light from the laser illuminates a small portion of the hologram, and the diffracted light converges at a viewing plane reconstructing a single perspective of the original object.
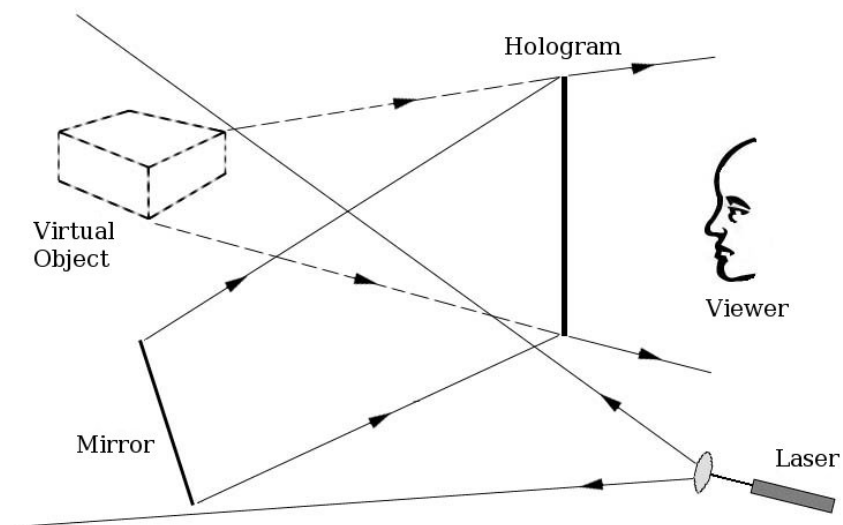

Figure 2.4: Typical optical hologram virtual image reconstruction. The light reflected from the mirror passes through the hologram, and appears to diverge from the location of the original object.

Chapter 3

Computational Holography

As previously discussed, the two major steps in optical holography involve the creation of the holographic fringe pattern and the reconstruction of the object from the hologram. Either of these steps can be done optically or computationally. Table 3.1 shows applications of holography in which the different combinations have been performed. The term computational holography refers to the use of digital representations and computational methods to either construct or reconstruct a hologram. The term digital holography is typically reserved for methods in which a hologram is optically constructed, digitally recorded (by way of a charge coupled device, for instance), then numerically reconstructed. We will use the term computer generated hologram (CGH) to refer to any holographic fringe pattern generated computationally.

Table 3.1: Example applications of computational holography.

Construction Technique	Reconstruction Technique	Examples
optical	optical	traditional film holography
optical	computational	digital holography [JFTC01], microscopy [YKOM01], tomography [GAD$^+$02, MRM$^+$05, Pir02, SR00]
computational	optical	traditional CGHs
computational	computational	holographic encryption [TJ00, WZJW06]

The first CGH was created by Lohmann and Paris, and was limited to small 2D images [LP67]. With computational abilities at hand, it is possible to create holograms of just about anything. The reconstruction of an arbitrary 3D wavefront is a powerful tool, and applications in any field of visualization become immediately apparent. The past 40 years has seen gradual improvements in the quality and capabilities of CGHs, yet there is much more to be done.

Each step in the holographic process presents its own problems. These stem from the fact that holography relies on diffraction and diffraction relies on very small apertures, on the order of the wavelength of the light being used. Because of this the shear amount of data to be computationally processed grows very quickly. As an example of this growth, let us look at finding the information content of a holographic fringe pattern from a sampling viewpoint [SHBL92]. The grating equation states that

$$f\lambda = \sin(\theta)$$

where f is the spatial frequency of the fringes, λ is the wavelength of light, and θ is the angular width of the diffracted light. The Nyquist sampling theorem states that the required minimum sampling rate for the hologram should be twice that of the spatial frequency [BH93], thus

$$f_{nyquist} = 2f.$$

Therefore a one-dimensional hologram of width w would require N samples, where

$$N = w f_{nyquist} = \frac{2w \sin(\theta)}{\lambda}$$

and a two-dimensional hologram of width w and height h would require M samples, where

$$M = w h f_{nyquist}^2 = \frac{4wh \sin^2(\theta)}{\lambda^2}.$$

Now assume that we produce a holographic display comparable to current two-dimensional displays, say 20cm by 15cm, with a viewing angle of 45°. Using a wavelength of 633nm, would require approximately 1.5×10^{11} samples of information. Assuming we used one byte of information per sample and wanted to generate 30 frames per second, this would require a data rate of approximately 4.5 TB/s. This data rate only considers the final hologram for display, and not the additional computational time that would be required for the calculation of the fringe pattern.

Eventually, a real-time holographic device having this capacity will exist, but for now we are severely limited by current technology. This chapter will explore the existing techniques in the computational construction and reconstruction of holograms. First, we will cover the necessary mathematical concepts and notations used throughout this dissertation, including complex mathematics and the Fourier transform. Then we will discuss how the physics of holography is modeled computationally, after which current techniques in computational holography will be discussed.

3.1 Complex Math

The chief concern of holography is the propagation and interference of wave-like signals. The principal mechanism that allows for the construction and reconstruction of holograms is the preservation of both the amplitude and phase of the signal being used. The use of complex mathematics will prove to be useful in the modeling of such systems.

Let z be a complex number. Then,

$$z = Re\{z\} + i\, Im\{z\}$$

where $Re\{z\}$ is the real part, $Im\{z\}$ is the imaginary part, and $i = \sqrt{-1}$. The number z can be viewed as a 2-element vector in real-imaginary space, having a component along the real-axis of $Re\{z\}$ and a component along the imaginary-axis of $Im\{z\}$, as shown on Figure 3.1.

Any complex number z can also be written as

$$z = \|z\|\, e^{i\phi}$$

where

$$\|z\| = \sqrt{Re^2\{z\} + Im^2\{z\}}$$

and

$$\phi = \tan^{-1}\left[\frac{Im\{z\}}{Re\{z\}}\right].$$

9
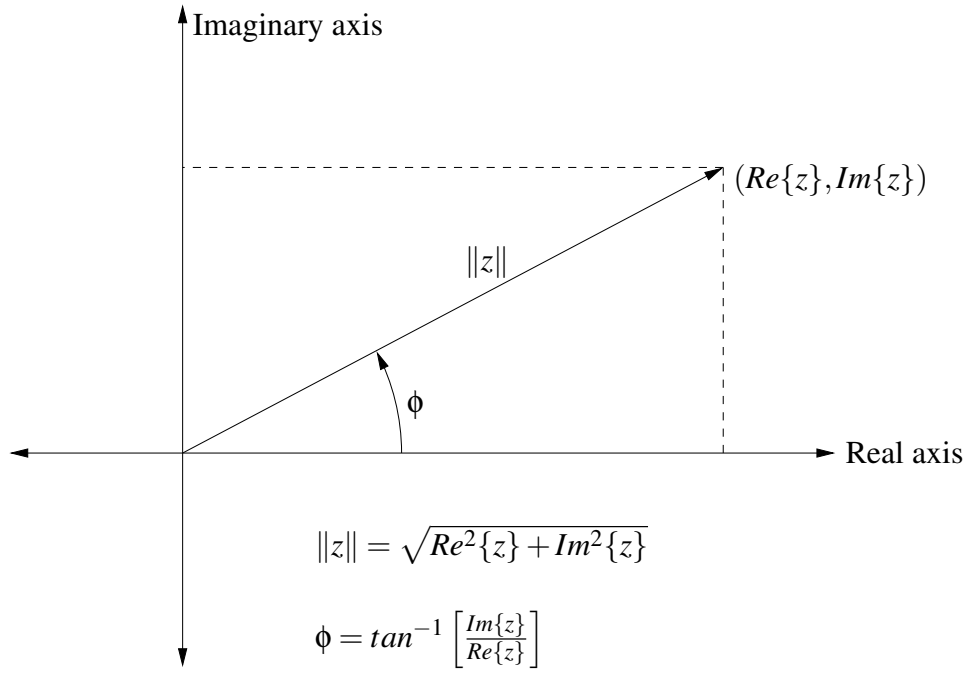
Figure 3.1: Vector representation of a complex number.

The quantity $\|z\|$ is the amplitude or modulus, and ϕ is the phase. We can see that $\|z\|$ is the magnitude of the vector representing z and ϕ is the angle the vector makes with the positive Real axis. This exponential complex notation is used extensively in this dissertation, because it simplifies the algebra involved. The following identity is useful in converting between the two notations, and is known either as the Euler formula or Euler Identity:

$$\|z\|\, e^{\pm i\phi} = \|z\| \cos(\phi) \,\pm\, i\, \|z\| \sin(\phi).$$

The addition of two complex numbers, say z_1 and z_2, consists of adding the real and imaginary parts separately:

$$z = z_1 + z_2 = (Re\{z_1\} + Re\{z_2\}) + i\,(Im\{z_1\} + Im\{z_2\}).$$

The multiplication of two complex numbers utilizes the distributive law and the fact that $i^2 = -1$. A general example of complex multiplication is:

$$\begin{aligned}
z = z_1\,z_2 &= (Re\{z_1\}\,Re\{z_2\} + i\,Im\{z_1\}\,i\,Im\{z_2\}) \\
&\quad + (Re\{z_1\}\,i\,Im\{z_2\} + i\,Im\{z_1\}\,Re\{z_2\}) \\
&= (Re\{z_1\}\,Re\{z_2\} - Im\{z_1\}\,Im\{z_2\}) \\
&\quad + i\,(Re\{z_1\}\,Im\{z_2\} + Im\{z_1\}\,Re\{z_2\}).
\end{aligned}$$

Many times we will only be interested in the amplitude of some resulting expression. For instance, in simulating the reconstruction of a hologram we should remember that our eyes only detect the amplitude of light and are not sensitive to its phase. Thus the reconstruction is independent of any multiplicative phase factors. For example, let $z = z_1\,z_2$ where $z_1 = a + i\,b$ and $z_2 = e^{i\phi}$ is simply a phase factor. Making use of the Euler Identity, we have:

$$\begin{aligned}
z &= (a + i\,b)(\cos(\phi) + i\,\sin(\phi)) \\
&= [a\cos(\phi) - b\sin(\phi)] + i\,[a\sin(\phi) + b\cos(\phi)]
\end{aligned}$$

10

Again, if we are only interested in the amplitude of z, we see that:

$$
\begin{aligned}
\|z\| &= \sqrt{(a\cos(\phi) - b\sin(\phi))^2 + (a\sin(\phi) + b\cos(\phi))^2} \\
&= \sqrt{a^2\cos^2(\phi) + b^2\sin^2(\phi) - 2ab\cos(\phi)\sin(\phi) + a^2\sin^2(\phi) + b^2\cos^2(\phi) + 2ab\cos(\phi)\sin(\phi)} \\
&= \sqrt{a^2(\sin^2(\phi) + \cos^2(\phi)) + b^2(\sin^2(\phi) + \cos^2(\phi))} \\
&= \sqrt{a^2 + b^2} \\
&= \|z_1\|
\end{aligned}
$$

The irradiance, or power function, of z is defined as:

$$
\begin{aligned}
\|z\|^2 &= Re^2\{z\} + Im^2\{z\} \\
&= z\,z^*
\end{aligned}
$$

where $z^* = Re\{z\} - i\,Im\{z\}$ denotes the complex conjugate of z.

3.2 The Fourier Transform

Unfortunately there are many conflicting notational definitions of a Fourier transform in use today. More importantly, though, they all have the same underlying meaning. Given some physical process, its representation can be described using some function f through time, in which we say is in the time domain. Yet it can also be described equivalently in the frequency domain, in which we have some related function F giving the value of the process at each frequency. These two approaches should be seen as two different representations of the same physical function. Thus given the values in one domain, we are able to transform into the other domain. This is the basis for the formulation of the Fourier transform. It allows us to transform between different yet equivalent representations of a signal. This explanation using a temporal (time-domain) signal extends to spatial signals. Any distinction between the two is semantics; the mathematical formulations are the same. Optical Holography is concerned with spatial signals, or simply, images. Holography relies on the ability to capture the spatial frequencies of these images, which present themselves as the interference of light waves, and to reconstruct the original image from those frequencies.

Many fields have somewhat abandoned the Fourier transform in favor of more robust signal processing tools such as Wavelets. Wavelets allow one to not only know what frequencies are present but approximately where they are present, either spatially or temporally. But with holography, there is a different need. We need the frequency information to be spread throughout the entire hologram; it is a necessity instead of a hindrance.

3.2.1 The Continuous Fourier Transform

The one-dimensional Fourier transform of a function $f(x)$ is a function of the frequency variable ξ and is defined as:

$$
\mathcal{F}[f] = F(\xi) = \int_{-\infty}^{\infty} f(x)\, e^{-i2\pi\xi x}\, dx. \tag{3.1}
$$

In order to reconstruct the original function from its frequency representation, we perform the inverse Fourier transform:

$$\mathcal{F}^{-1}[F] = f(x) = \int\limits_{-\infty}^{\infty} F(\xi)\, e^{i2\pi\xi x}\, d\xi.$$

These two functions form what is called a Fourier transform pair, such that $f = \mathcal{F}^{-1}[\mathcal{F}[f]]$ and $F = \mathcal{F}[\mathcal{F}^{-1}[F]]$. The generalization to two dimensions is straightforward:

$$F(\xi,\eta) = \int\limits_{-\infty}^{\infty}\int\limits_{-\infty}^{\infty} f(x,y)\, e^{-i2\pi(\xi x+\eta y)}\, dxdy$$

$$f(x,y) = \int\limits_{-\infty}^{\infty}\int\limits_{-\infty}^{\infty} F(\xi,\eta)\, e^{i2\pi(\xi x+\eta y)}\, d\xi d\eta.$$

If the function f is separable into two functions, namely

$$f(x,y) = g(x)\, h(y),$$

then the Fourier transform of f is the product of the Fourier transforms of g and h:

$$\begin{aligned}
F(\xi,\eta) &= \int\limits_{-\infty}^{\infty}\int\limits_{-\infty}^{\infty} f(x,y)\, e^{-i2\pi(\xi x+\eta y)}\, dxdy \\
&= \int\limits_{-\infty}^{\infty}\int\limits_{-\infty}^{\infty} g(x)\, h(y)\, e^{-i2\pi(\xi x+\eta y)}\, dxdy \\
&= \int\limits_{-\infty}^{\infty} g(x)\, e^{-i2\pi\xi x}\, dx \int\limits_{-\infty}^{\infty} h(y)\, e^{-i2\pi\eta y}\, dy \\
&= G(\xi)\, H(\eta).
\end{aligned}$$

We will call a transform of the following form:

$$S_s[f] = S_s(\xi) = \int\limits_{-\infty}^{\infty} f(x)\, e^{-i2\pi s\xi x}\, dx \tag{3.2}$$

a scaled Fourier transform, denoted by the operator S and subscripted with the scale parameter s. To see the reason for this choice of terminology, let $\xi' = s\xi$. Equation 3.2 then becomes:

$$S_s(\xi) = F(\xi') = \int\limits_{-\infty}^{\infty} f(x)\, e^{-i2\pi\xi' x}\, dx, \tag{3.3}$$

which we see as the Fourier transform of our new frequency variable ξ'. For the extension to a two-dimensional scaled Fourier transform, we will allow a different scale parameter in each dimension. We will define such a transform as:

$$S_{s,t}[f] = S_{s,t}(\xi,\eta) = \int\limits_{-\infty}^{\infty}\int\limits_{-\infty}^{\infty} f(x,y)\, e^{-i2\pi(s\xi x+t\eta y)}\, dx\, dy. \tag{3.4}$$

3.2.2 The Discrete Fourier Transform

The discrete approximation to the continuous Fourier transform begins as many discretization problems do: with sampling. We begin by sampling a continuous function f at N values of x at regular intervals of Δ [PTVF92]. It is assumed that our function is nonzero only within our sampled interval. Our continuous function f is then approximated by a discrete indexed function

$$f_j = f(j\Delta) , \quad j = 0, 1, 2, \ldots, N-1 .$$

The discrete frequency values will be calculated at

$$v_k = \frac{k}{N\Delta} , \quad k = -\frac{N}{2}, \cdots, \frac{N}{2} .$$

The continuous Fourier transform can then be approximated as [PTVF92]:

$$F(\xi) = \int_{-\infty}^{\infty} f(x)\, e^{-i2\pi\xi x}\, dx \approx \sum_{j=0}^{N-1} \Delta\, f_j\, e^{-i2\pi v_k j\Delta} = \Delta \sum_{j=0}^{N-1} f_j\, e^{-i2\pi jk/N} . \tag{3.5}$$

A direct implementation of the discrete Fourier transform of N sampled points would require on the order of N^2 operations; it has $O(N^2)$ complexity. Possibly one of the most valuable and used algorithms in signal processing is the fast Fourier transform (FFT). The FFT allows for the calculation of the one-dimensional discrete Fourier transform in $O(N\log N)$ time, meaning that the number of operations is on the order of $N\log N$ (where the base of the logarithm is 2). The complexity for a two-dimensional FFT is $O(N^2\log N)$. This results in significant savings in computational time for large N. Many independent derivations of the FFT have been discovered and rediscovered [HJB85]. The first widely used algorithms were based on a formulation by J.W. Cooley and J.W. Tukey [CT65]. The FFT algorithm used in this research was adapted from the Numerical Recipes text [PTVF92].

Section 3.2.1 defined a scaled Fourier transform. An obvious way to digitally compute Equation 3.2 would involve defining $\xi' = s\xi$ and sampling this new frequency variable. This can unfortunately lead to extreme aliasing issues when sampling in the frequency domain is too low. Bailey and Swarztrauber derived an alternative method of computing a scaled Fourier transform [BS91]. We will not refer to their definition as a fractional Fourier transform, as they did, to avoid confusion with the more popular definition of a fractional Fourier transform discussed later in Section 3.2.3.

The discretization of the scaled Fourier transform given by Bailey and Swarztrauber is similar to the discrete Fourier transform in Equation 3.5 [BS91]:

$$G_k = \sum_{j=0}^{N-1} f_j\, e^{-i2\pi s jk} , \tag{3.6}$$

for some complex number s. Noting that $2jk = j^2 + k^2 - (k-j)^2$, Equation 3.6 can be expressed as:

$$\begin{aligned} G_k &= \sum_{j=0}^{N-1} f_j\, e^{-i\pi s[j^2 + k^2 - (k-j)^2]} \\ &= e^{-i\pi s k^2} \sum_{j=0}^{N-1} f_j\, e^{-i\pi s j^2}\, e^{i\pi s(k-j)^2} \\ &= e^{-i\pi s k^2} \sum_{j=0}^{N-1} y_j\, z_{k-j} , \end{aligned} \tag{3.7}$$

13

where y_j and z_j are defined as:

$$y_j = f_j \, e^{-i\pi s j^2}$$
$$z_j = e^{i\pi s j^2} \, .$$

(3.8)

The final summation in Equation 3.7 is in the form of a convolution. The convolution between two functions, $y(x)$ and $z(x)$, is defined as [BJF70]

$$g(x) = y(x) * z(x) = (y * z)(x) = \int_{-\infty}^{\infty} y(\tau) \, z(x - \tau) \, d\tau,$$

(3.9)

where the asterisk is used to denote convolution. The Convolution Theorem links this definition with the Fourier transform, stating that the Fourier transform of the convolution between two functions is equal to the product of the Fourier transforms of the same two functions. Thus can be explicitly stated as:

$$\mathcal{F}[g] = \mathcal{F}[y * z] = \mathcal{F}[y] \, \mathcal{F}[z] \, .$$

(3.10)

Making use of the inverse Fourier transform, g can be written as:

$$g = \mathcal{F}^{-1}[\, \mathcal{F}[y] \, \mathcal{F}[z] \,] \, .$$

(3.11)

The convolution in this form facilitates fast computation with the FFT algorithm. Therefore Equation 3.7, the discrete scaled Fourier transform, can be computed efficiently with the Convolution Theorem and the FFT algorithm. We will refer to this method as the fast scaled Fourier transform.

3.2.3   The Fractional Fourier Transform

It has long been known that the hologram of an image at infinity, known as a Fourier hologram, is proportional to the Fourier transform of that image [Har96]. To carry this process out optically, a lens of focal length f is typically used. The lens is located a distance f from the hologram and the image is placed at the focal point of the lens, a distance $2f$ from the hologram. During reconstruction of the hologram, the propagating wavefront oscillates between reproducing the image, an inverted hologram, an inverted image, the hologram, the image, etc., as the distance from the hologram increases [MO93, OM93]. It seems natural to think that this propagating wavefront makes a continuous change while transforming between each representation. Only recently has it been shown that fractional powers of the Fourier transform describe this propagating wavefront [MO93, Loh93, OM93, OBMO94, MZD$^+$95]. There are many accepted derivations of the fractional Fourier transform [OZK01]. In this research we follow its definition as a linear integral transform.

The one-dimensional fractional Fourier transform of a function $f(x)$ of order $a \neq 2j$ (j is some integer) is a function of the frequency variable ξ and is defined as [OZK01]:

$$\mathcal{F}^a[f] \equiv F_a(\xi) \equiv \int_{-\infty}^{\infty} f(x) \, A_\alpha \, e^{i\pi[\cot(\alpha)\xi^2 - 2\csc(\alpha) x\xi + \cot(\alpha) x^2]} \, dx$$

(3.12)

where

$$\alpha \equiv \frac{a\pi}{2},$$
$$A_\alpha \equiv \sqrt{1 - i\cot(\alpha)} \, .$$

(3.13)

14

The fractional Fourier transform produces the original function when $a = 0$ or $a = 4j$, and produces the parity function, $f(-x)$, when $a = 4j \pm 2$. If we constrain a such that $0 < |a| < 2$, then A_α can be written as [OZK01]:

$$A_\alpha = \frac{e^{-i[\pi \operatorname{sgn}(\alpha)/4 - \alpha/2]}}{\sqrt{|\sin(\alpha)|}}.$$

If $|a| \geq 2$, then we simply use $a \bmod 4$ — the result of the transform is the same because of its periodic nature. As stated earlier, the fractional Fourier transform represents a generalization of the more familiar Fourier transform. To show this, we choose $a = 1$. Thus $\alpha = \pi/2$, $A_\alpha = 1$, and the transform becomes:

$$F_1(\xi) = \int_{-\infty}^{\infty} f(x) \, e^{-i2\pi x \xi} \, dx,$$

which is the traditional Fourier transform from Equation 3.1. An important property of the fractional Fourier transform is its index additivity. The result of a transform of order a_1 followed by a transform of order a_2 is equivalent to a single transform of order $a_1 + a_2$. This is stated symbolically as:

$$\mathcal{F}^{a_1} \mathcal{F}^{a_2} = \mathcal{F}^{a_1 + a_2} = \mathcal{F}^{a_2} \mathcal{F}^{a_1}. \tag{3.14}$$

As will be explored in Chapter 4, the propagated diffraction pattern of a hologram is proportional to a fractional Fourier transform.

3.2.4 A Fast Fractional Fourier Transform

An efficient implementation of an approximate fractional Fourier transform can be reached by rearranging Equation 3.12 as [OZK01]:

$$F_a(\xi) = A_\alpha \, e^{i\pi \cot(\alpha) \xi^2} \int_{-\infty}^{\infty} \left[f(x) e^{i\pi \cot(\alpha) x^2} \right] e^{-i2\pi \csc(\alpha) x \xi} dx. \tag{3.15}$$

The expression within brackets is computed first, followed by a scaled Fourier transform (see Section 3.2.2), and finally by multiplication of the leading A_α and exponential terms. The overall complexity for this calculation is $O(N \log N)$. Numerical problems arise in the sampling of the exponential functions when a is near an even integer [OZK01]. This issue is dealt with by utilizing the periodic nature of the transform as well as its index additivity property.

3.3 Modeling Light, Diffraction, and Interference

Let U be a three-dimensional scalar function defined throughout time. If we say U is a wave function, then it satisfies the wave equation [Fow89]

$$\nabla^2 U = \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = \frac{1}{v^2} \frac{\partial^2 U}{\partial t^2} \tag{3.16}$$

where v is the velocity of the wave. The specifics of this equation for the interested reader can be found in [Gri89] for a more in-depth study of its origins and consequences. The reason for using this equation is that light is an electromagnetic wave which satisfies the wave equation. Thus we will use functions in our modeling of holography which satisfy the wave equation. Knowledge and understanding of the amplitude and phase of a light wave through time and space are required for proper modeling.

The sinusoidal functions are perhaps the most familiar wave form, and it will be the chosen function for modeling the behavior of light. For now we restrict ourselves to one-dimensional functions — later extension into two or three dimensions will be given. Let f be a function defined as:

$$f(x,t) = A_0 \cos(kx - \omega t)$$
$$\text{where } k = \frac{2\pi}{\lambda}, \omega = kv. \tag{3.17}$$

The wavelength is the distance between successive valleys or troughs of the cosine function, and is represented by λ. The amplitude of the wave, designated by A_0, is the absolute value of the highest and lowest values of the function. The quantity $k = 2\pi/\lambda$ is called the wavenumber, and v is the velocity of the wave. Equation 3.17 can be shown to satisfy the wave equation by direct substitution into Equation 3.16. This formulation represents what is known as a plane harmonic wave. Since the algebraic manipulation of exponentials can be simpler than that of cosine or sine functions, note that:

$$f(x,t) = Re\{A_0\, e^{i(kx - \omega t)}\}.$$

As is common in computational holography, we will use the exponential form in modeling light waves. It should be understood that the real part of the complex number represents the actual physical value. It should also be noted that the complex representation satisfies the wave equation.

We will now extend this definition to three-dimensions [Fow89]. Let us define the position vector $\mathbf{r}$ as:

$$\mathbf{r} = x\hat{\mathbf{i}} + y\hat{\mathbf{j}} + z\hat{\mathbf{k}},$$

where $\hat{\mathbf{i}}, \hat{\mathbf{j}}$, and $\hat{\mathbf{k}}$ are the familiar unit vectors. The propagation vector, given by

$$\mathbf{k} = k_x\hat{\mathbf{i}} + k_y\hat{\mathbf{j}} + k_z\hat{\mathbf{k}},$$

is defined such that its magnitude is equal to the wavenumber:

$$|\mathbf{k}| = \frac{2\pi}{\lambda} = \sqrt{k_x^2 + k_y^2 + k_z^2}.$$

Noting that the dot product of $\mathbf{k}$ and $\mathbf{r}$ is

$$\mathbf{k} \cdot \mathbf{r} = k_x x + k_y y + k_z z,$$

the three-dimensional plane harmonic function becomes:

$$f(x,y,z) = A_0\, e^{i(k_x x + k_y y + k_z z - \omega t)} = A_0\, e^{i(\mathbf{k} \cdot \mathbf{r} - \omega t)}.$$

These are plane waves, having a single direction of propagation. They have constant phase on planes normal to the propagation vector $\mathbf{k}$. Spherical waves will be used later in the modeling of diffraction

and interference. These waves have constant phase on spheres of radius r centered on their source at any time t. In order to satisfy the wave equation, spherical waves will be written as [Gri89, Fow89]:

$$f(r,t) = A_0 \frac{e^{i(kr-\omega t)}}{r} \tag{3.18}$$

where k and r are again scalar values.

In this dissertation, it is assumed that we are always using a monochromatic (single wavelength) coherent (same initial phase) light source. When addressing the interference of two wavefronts, we are only concerned about the relative difference in their phase across some spatial region. Since all the light waves have the same velocity, they all have the same value for ω, and thus the ωt term in the exponential can be factored. For instance, the addition of two wavefronts can be written as:

$$A_1 e^{i(kx_1-\omega t)} + A_2 e^{i(kx_2-\omega t)} = e^{-i\omega t} \left[A_1 e^{ikx_1} + A_2 e^{ikx_2} \right].$$

Since we are only concerned with the relative differences arising between the interference of wavefronts, we are free to choose the value of t. Thus let us choose $t = 0$, so that the leading exponential term becomes 1. Noting that the time dependence has been removed, the plane harmonic function in one-dimension becomes:

$$f(x) = A_0 e^{ikx}.$$

Let us turn our attention to the process of simulating diffraction. Huygen's Principle essentially states that the progression of a wavefront can be determined by assuming that there are infinitely many point sources on the wavefront, each emitting a spherically expanding wave. Thus the new wavefront is understood to be the envelope of this secondary set of expanding waves [Fow89]. This is simply an intuitive description of the physics involved, but it ends up producing a valid approximation. For instance, calculating the diffracted light from a square aperture would involve summing the contributions from an infinite number of point sources located in the square each emitting a spherically propagating wave of light.

This principle can be described mathematically using the Fresnel-Kirchhoff integral formula. Figure 3.2 illustrates the principal optical setup (based on a Figure from [Fow89]). The aperture is a transparent area of space surrounded by an opaque barrier. The point S represents a point source of light from which we wish to determine the form of the light reaching point P. The vectors $\mathbf{r}$ and $\mathbf{r}'$ specify the positions of points P and S, respectively, and the vector $\mathbf{n}$ specifies the normal to the aperture. The Fresnel-Kirchhoff formula [Fow89, Joo86] can be stated as:

$$U_P = -\frac{ikU_0 e^{-i\omega t}}{4\pi} \int\limits_A \frac{e^{ik(r+r')}}{rr'} \left[\cos(\mathbf{n} \cdot \mathbf{r}) - \cos(\mathbf{n} \cdot \mathbf{r}') \right] dA \tag{3.19}$$

where the integral is calculated over the aperture. This represents the most general scalar diffraction formula. For this research, the value of the wavefront at the aperture will always be given by some function, $g(x)$. This function replaces the terms in the Fresnel-Kirchhoff formula describing the wavefront at the aperture from S, namely:

$$U_0 \frac{e^{ikr'}}{r'}.$$

We may also make the assumptions that the obliquity factor, $[\cos(\mathbf{n} \cdot \mathbf{r}) - \cos(\mathbf{n} \cdot \mathbf{r}')]$, is approximately 2 and that $t = 0$ so that $e^{-i\omega t} = 1$ [Fow89]. Therefore the integral formula for diffraction becomes:

$$U_P = -\frac{ik}{2\pi} \int\limits_{-\infty}^{\infty} g(x) \frac{e^{ikr}}{r} dx, \tag{3.20}$$

17
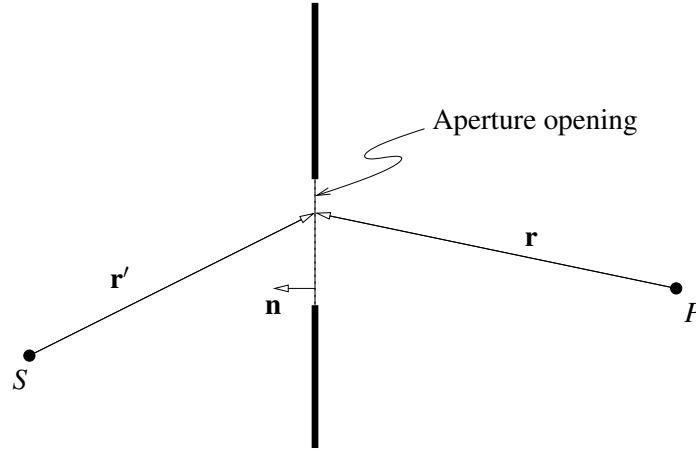
Figure 3.2: Fresnel-Kirchhoff formula configuration.

where r is the distance from each point on the aperture to the point P, and is a function of x. Comparing Equation 3.20 with Equation 3.18, we see how the diffracted wavefront is computed by summing over the aperture as if it were comprised of point sources each emitting a spherical wave. In the modeling of interference, we will further simplify Equation 3.20 by dropping the term $-ik/2\pi$, which does not effect the interference.

For the purposes of holographic interference we are typically interested in the interference at some two-dimensional plane, for instance the plane corresponding to the holographic film. Let $U(x,y)$ denote the complex wavefront at the x-y plane resulting from the object. This could be a result of light scattered from the object or light emitted by the object, and represents the self-interference of the object points. At any point on the plane, this wavefront has both an amplitude defined by $a(x,y)$ and a phase defined by $\phi(x,y)$. Thus we write $U(x,y)$ as:

$$U(x,y) = a(x,y)\,e^{i\phi(x,y)}.$$

Now suppose that a reference wave, U_0, is incident upon the x-y plane. It can similarly be defined as:

$$U_0(x,y) = a_0(x,y)\,e^{i\phi_0(x,y)}.$$

These two wavefronts will interfere at the x-y plane and produce a standing wave pattern. The irradiance produced by this interference is:

$$I(x,y) = \|U + U_0\|^2.$$

Since U and U_0 are both complex, this becomes:

$$
\begin{aligned}
I(x,y) &= \|U\|^2 + \|U_0\|^2 + UU_0^* + U^*U_0 \\
&= a^2 + a_0^2 + UU_0^* + U^*U_0 \\
&= a^2 + a_0^2 + aa_0\,e^{i(\phi-\phi_0)} + aa_0\,e^{-i(\phi-\phi_0)} \\
&= a^2 + a_0^2 + 2aa_0\cos(\phi - \phi_0).
\end{aligned}
$$

In optical holography, this interference is recorded by the film in such as way that transmittance of light through the developed film would be proportional to this irradiance value. Suppose then that a similar

18

reference beam is incident upon the hologram. Aside from a constant value, the transmitted light would be:

$$U_T(x,y) = IU_0$$
$$= a^2 U_0 + a_0{}^2 U_0 + U U_0^* U_0 + U^* U_0 U_0$$
$$= (a^2 + a_0{}^2) U_0 + a_0{}^2 U + U^* U_0{}^2.$$

The first term, $(a^2 + a_0{}^2)U_0$, is the directly transmitted beam proportional to the reference beam. The second term, $a_0{}^2 U$, is a wavefront directly proportional to the original object wavefront, which constructs the virtual image. The final term, $U^* U_0{}^2$, creates the real image.

3.4 CGH Construction Techniques

As the example of the amount of information present in a hologram (page 8), the data to be computed for a hologram of comparable size to two-dimensional displays can be intimidating. Various approaches at CGH construction have been explored since the first ones were created by Lohmann and Paris [LP67]. Typically a digital representation of the desired object is created, a numerical simulation of optical interference in some form is carried out, and the holographic fringe pattern is calculated. We will briefly discuss three important approaches to CGH construction: interference-based, Fourier-based, and diffraction-specific.

3.4.1 Interference-based Hologram Computation

The interference-based approach to hologram construction closely resembles the actual physical process of optical holography, more so than other approaches. The object in question is viewed as being made up of point sources of light, where each point emits a spherical wavefront. In most instances of interference-based fringe computation, every point in the hologram samples every point of the object. With the need for high resolution holograms (to allow for diffraction and the necessary parallax) and the desire for high resolution objects, it is easy to see why computational times are so large. The Fresnel-Kirchhoff formula (Equation 3.19) is generally applied directly, integrating over a point-based object.

The diffraction of geometric primitives other than points has been studied [RBD$^+$97, WkFB99, MSW02, MK03]. In these approaches, objects are made up of lines, planes, and/or triangles and the holographic fringe pattern is computed from a summation of the diffraction patterns from each primitive. The main advantage of interference-based computation arises in its similarity to ray-tracing algorithms of 3D rendering to two dimensions.

3.4.2 Fourier-based Hologram Computation

Fourier-based hologram computation refers to approaches that take advantage of the similarity of diffraction formulas to Fourier transforms. By proper manipulation, and under certain assumptions, the Fresnel-Kirchhoff formula (Equation 3.19) can be cast into a form which resembles the Fourier transform (Equation 3.1), thus allowing for fast computation using the FFT (see Section 3.2.2).

The main disadvantage of this approach is that the hologram generation of three-dimensional objects is difficult. Objects are typically divided into slices perpendicular to the hologram plane, each Fourier transformed, then summed to give the final hologram. Even utilizing fast algorithms, this can be wasteful compared to interference-based methods when considering sparse objects. Also, each slice being transformed separately does not allow for proper occlusion effects to be considered (where objects in the foreground block objects in its background).

Nevertheless, Fourier-based methods maintain a distinct advantage because they are constructed with fast Fourier transforms (see Section 3.2.2).

3.4.3 Diffraction-specific Hologram Computation

The desire to avoid problems associated with interference-based approaches led to the development of an alternate technique called diffraction-specific computation [Luc94, Luc96]. In this approach, the reconstruction step is considered directly. The act of simulating the interference is avoided altogether. A large set of basis fringes having specific diffractive properties are precomputed and stored. Given an object, the necessary diffractive behavior of each region of the hologram is determined and computed from the stored basis fringes using linear superpositioning.

3.5 CGH Reconstruction Techniques

A traditional optical hologram is developed using film, and early computational holography mimicked this. The first optical CGH reconstructions consisted of printing the hologram at a larger scale using a plotter, then photographically reducing it to a scale suitable for diffraction [IIS71, Les89, Mac92, Tre95]. While this approach proved the validity of the holograms computed, it would never lend itself to real-time holography.

Various electronically controlled spatial light modulators (SLM) have been used to reconstruct holograms, including transmission type liquid crystal displays (LCD), reflective LCD (or liquid crystal on silicon) [IS04, SSMI05], acoustic optic modulators (AOM) [Luc94, Que04, PAIK05], and digital micromirror devices (DMD) [HMG03, HMHG05]. With the exception of the AOM, these devices allow for the control of two-dimensional discrete grids of microscopic apertures. Because of the small size of these apertures, diffraction of coherent light can be produced that is capable of reconstructing holograms. An AOM consists of an ultrasonic transducer sending shear waves through a crystal, controlled by an electrical signal. The index of refraction changes based on this shear wave, and thus the diffractive properties can be matched to the desired holographic fringe pattern [SHBL92].

The major drawback with these systems is large modulation elements that limit the amount of diffraction that can be produced, thus limiting the available viewing angle [LCR02]. Typical aperture sizes range from 10 microns to 16 microns [HMG03, IS04, SSMI05]. This will likely become less of an issue as the technology of micro-fabrication increases and once a profitable demand for such devices exists.

The previously mentioned SLMs were generally not designed with holography in mind, e.g. the Texas Instruments' DMD. The DMD, one type of Digital Light Processing (DLP) chip, consists of a 2D array of mirrors where each mirror is capable of tilting independently in one of two directions directed by an

electrical signal. Traditional applications for the chip use the direct reflection, or zero order, of white light for 2D imaging. Grayscale is obtained by modulating the amount of time that each mirror is in the "on" position, referred to as pulse width modulation. The reflectance from each mirror is averaged over time by the human visual system, thus producing a grayscale value for white light applications.

Huebschman, et al., recently developed a means of reconstructing holograms by using DMD chips removed from high-definition video projectors [HMG03, HMHG05]. Huebschman's group applied an interference-based approach for the holographic fringe calculation. The intensity for each pixel on the hologram was derived by integrating over the desired object, similar to ray tracing. They discovered that binary holograms reconstructed better than grayscale. This is because the DMD is in essence a binary device. It is incapable of producing a grayscale pattern in a single instant. Trying to reconstruct a grayscale hologram on a DMD produces a series of binary holograms, each of which poorly reconstructs the image and is averaged over time by the human visual system.

In this research, the actual physical hologram reconstruction is pursued to demonstrate our new results. Existing computational methods as well as those developed here are used to digitally reconstruct the holograms. The techniques developed here are essentially independent of the specific reconstruction device being used.

Chapter 4

Fast Computation of Spatially Shifted Diffraction

As mentioned in Section 3.2.2, a direct implementation of the one-dimensional discrete Fourier transform has complexity $O(N^2)$ while the one-dimensional FFT has complexity $O(N \log N)$. Also, the complexity of a direct implementation of the two-dimensional discrete Fourier transform has complexity $O(N^4)$ while the two-dimensional FFT has complexity $O(N^2 \log N)$. In this research, we will refer to algorithms having $O(N \log N)$ complexity in one dimension or $O(N^2 \log N)$ complexity in two dimensions as "fast", as opposed to slower implementations that result in $O(N^2)$ or $O(N^4)$ complexity.

In traditional Fourier-based holographic techniques (see Section 3.4.2), assumptions are made so that diffraction formulas have the form of a Fourier transform and the FFT can be used. This formulation, along with its restrictions, will be discussed in Section 4.1. Our purpose in this chapter is to derive fast algorithms that overcome some of the limitations of Fourier-based techniques.

4.1   Fresnel and Fraunhofer Diffraction

We begin the derivation of the Fresnel diffraction integral starting with our simplified Fresnel-Kirchhoff integral formula in Equation 3.20. Given a complex-valued wavefront of $\psi(x',y',z')$, we compute the propagated wavefront $\Psi(x,y,z)$. Using the modified Fresnel-Kirchhoff integral formula,

$$\Psi(x,y,z) = -\frac{ik}{2\pi} \int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} \psi(x',y',z') \, \frac{e^{ikr}}{r} \, dx' dy' dz', \tag{4.1}$$

where

$$r = \sqrt{(x'-x)^2 + (y'-y)^2 + (z'-z)^2},$$

we assume that our source wavefront is confined to a single plane and set $z' = 0$ and r simplifies to

$$r = \sqrt{(x'-x)^2 + (y'-y)^2 + z^2}.$$

Focusing on the kr term in the exponential function in Equation 4.1, we rearrange the terms to produce:

$$kr = k\sqrt{(x'-x)^2 + (y'-y)^2 + z^2}$$

$$= k\sqrt{z^2 \left[1 + \frac{(x'-x)^2}{z^2} + \frac{(y'-y)^2}{z^2} \right]}$$

$$= kz\sqrt{1 + \frac{(x'-x)^2}{z^2} + \frac{(y'-y)^2}{z^2}}$$

$$\approx kz \left[1 + \frac{(x'-x)^2}{2z^2} + \frac{(y'-y)^2}{2z^2} \right],$$

where the square root term is approximated by the first two terms of the power series [GR94]:

$$\sqrt{1+x} = 1 + \frac{x}{2} - \frac{x^2}{8} + \cdots .$$

Substitution into Equation 4.1 yields:

$$\Psi(x,y,z) = -\frac{ik}{2\pi} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi(x',y') \frac{1}{r} e^{ikz\left[1 + \frac{(x'-x)^2}{2z^2} + \frac{(y'-y)^2}{2z^2}\right]} dx'dy'$$

$$= -\frac{ik}{2\pi} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi(x',y') \frac{1}{r} e^{ikz} e^{i\frac{\pi}{\lambda z}\left[(x'-x)^2 + (y'-y)^2\right]} dx'dy' .$$

$$(4.2)$$

Assuming that $z = d$ is held constant and that $r \approx z$ leads to the Fresnel diffraction integral:

$$\Psi_d(x,y) = -\frac{ik}{2\pi}\frac{1}{d} e^{ikd} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi(x',y') e^{i\frac{\pi}{\lambda d}\left[(x'-x)^2 + (y'-y)^2\right]} dx'dy'$$

$$= \frac{1}{i\lambda d} e^{ikd} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi(x',y') e^{i\frac{\pi}{\lambda d}\left[(x'-x)^2 + (y'-y)^2\right]} dx'dy' .$$

$$(4.3)$$

The form of Equation 4.3, with the leading term written as $1/i\lambda d$, is commonly seen in the literature. The range of d for which the Fresnel diffraction integral is valid is given by [MGF+99]:

$$d \geq \frac{\Delta x^2 N + \Delta y^2 M}{\lambda}, \tag{4.4}$$

where Δx and Δy are the pitch in x and y directions, respectively. N is the resolution in the x direction and M is the resolution in the y direction. Equation 4.4 is the minimum Fresnel distance, and must be observed when applying the Fresnel diffraction integral.

Futher simplifications involving the exponential term in Equation 4.2 can be made when both $x^2 + y^2 \ll 2z^2$ and $x'^2 + y'^2 \ll 2z^2$. This results in the Fraunhofer diffraction integral [OO05]:

$$\Psi_d(x,y) = -\frac{ik}{2\pi}\frac{1}{d} e^{ikd} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi(x',y') e^{-i\frac{2\pi}{\lambda d}(x'x + y'y)} dx'dy' . \tag{4.5}$$

Fraunhofer diffraction is also known as far-field diffraction, since the object and hologram must both be small and be a large distance apart.

Traditional approaches that discretize the Fresnel or Fraunhofer diffraction integrals to obtain a fast algorithm restrict the sampled points to be centered about the axis of propagation and constrain the sampling rates involved. Presented next is a typical discretization of the Fresnel diffraction integral [MGF+99] — the discretization of the Fraunhofer diffraction is very similar.

We begin by rearranging Equation 4.3 into a more suitable form:

$$\Psi_d(x,y) = \frac{1}{i\lambda d} e^{ikd} e^{i\frac{\pi}{\lambda d}(x^2 + y^2)} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left[\psi(x',y') e^{i\frac{\pi}{\lambda d}(x'^2 + y'^2)}\right] e^{-i\frac{2\pi}{\lambda d}(x'x + y'y)} dx'dy' . \tag{4.6}$$

Let N be the sampled resolution in each dimension and in each domain, and let the sampling pitches be Δx, Δy, $\Delta x'$, and $\Delta y'$ in the x, y, x', and y' directions, respectively. Our sampled Fresnel transform becomes:

$$
\Psi_d(m\Delta x, n\Delta y) = \frac{1}{i\lambda d} e^{ikd} e^{i\frac{\pi}{\lambda d}\left[(m\Delta x)^2 + (n\Delta y)^2\right]} \times
$$
$$
\sum_p \sum_q \left[\psi(p\Delta x', q\Delta y') e^{i\frac{\pi}{\lambda d}\left[(p\Delta x')^2 + (q\Delta y')^2\right]} \right] e^{-i\frac{2\pi}{\lambda d}(p\Delta x' m\Delta x + q\Delta y' n\Delta y)}, \tag{4.7}
$$

where m, n, p, and q range from $-N/2 - 1$ to $N/2$. We impose the condition that

$$
\frac{\Delta x \Delta x'}{\lambda d} = \frac{\Delta y \Delta y'}{\lambda d} = \frac{1}{N}
$$

so that Equation 4.7 becomes

$$
\Psi_d(m\Delta x, n\Delta y) = \frac{1}{i\lambda d} e^{ikd} e^{i\frac{\pi}{\lambda d}\left[(m\Delta x)^2 + (n\Delta y)^2\right]} \times
$$
$$
\sum_p \sum_q \left[\psi(p\Delta x', q\Delta y') e^{i\frac{\pi}{\lambda d}\left[(p\Delta x')^2 + (q\Delta y')^2\right]} \right] e^{-i2\pi(pm+qn)/N}. \tag{4.8}
$$

The final exponential in Equation 4.8 is the kernel of the discrete Fourier transform (see Section 3.2.2). Thus computation of this value involves first multiplication of the input sampled function by the exponential in brackets in the summation, then calculation of the discrete Fourier transform using the FFT, and finally multiplication by the leading terms outside of the summation. Next we will derive an alternative discretization of the Fresnel diffraction integral that alleviates some of the restrictions discussed previously.

4.2   Shifted Fresnel Diffraction

The techniques developed in this section overcome the limitations of traditional Fourier-based holography by allowing for arbitrary spatial separation from the axis of propagation and arbitrary pitch values, while also permitting an efficient computational algorithm.

We begin by defining a finite rectangular cartesian grid in the x-y and x'-y' planes. Figure 4.1 illustrates the geometric setup described. The x'-y' plane is simply a shift of the x-y plane by a distance of d in the z direction. The x-y grid has one corner located at (x_0, y_0), with a resolution of $M \times N$, and has x and y pitch values of Δx and Δy, respectively. The x'-y' grid is defined similarly, having one corner located at (x'_0, y'_0), with a resolution of $P \times Q$, and has x' and y' pitch values of $\Delta x'$ and $\Delta y'$, respectively. The coordinates of the set of indexed points for these two grids are determined by the following formulas:

$$
\begin{aligned}
x_m &= x_0 + m\Delta x & x'_p &= x'_0 + p\Delta x' \\
y_n &= y_0 + n\Delta y & y'_q &= y'_0 + q\Delta y',
\end{aligned}
$$

where m, n, p, and q are integer indices with the following bounds:

$$
\begin{aligned}
0 &\leq m \leq M-1 & 0 &\leq p \leq P-1 \\
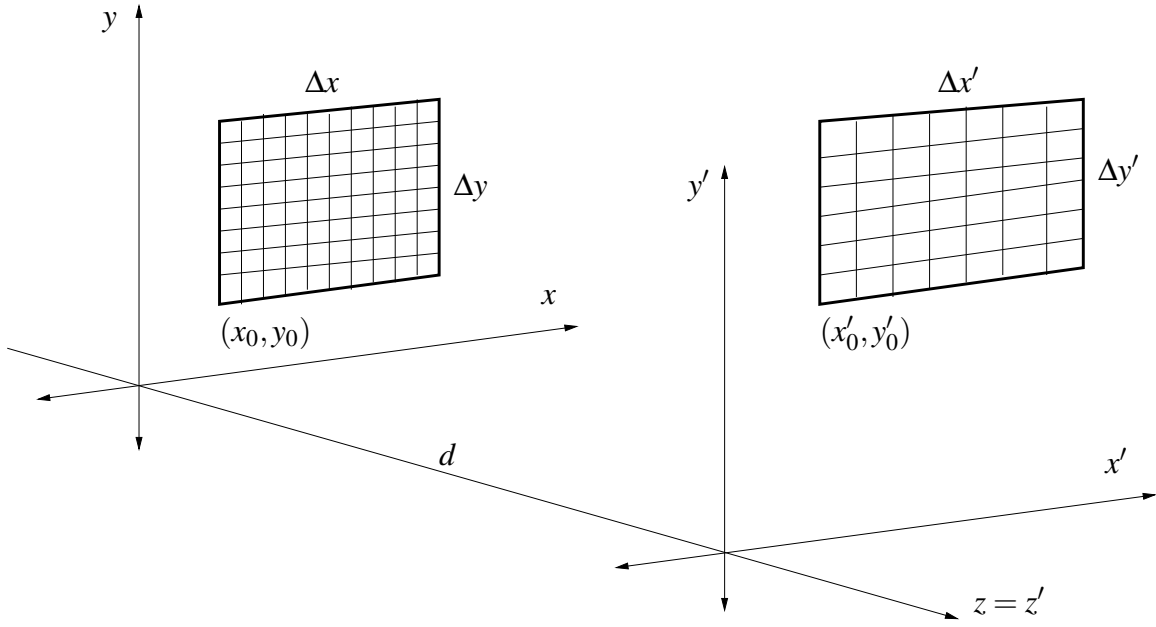0 &\leq n \leq N-1 & 0 &\leq q \leq Q-1.
\end{aligned}
$$

Figure 4.1: Shifted Fresnel diffraction geometry.

Using this discretization, we will recast Equation 4.6 by defining the following discrete functions that represent the source and target wavefronts:

$$U_d(m,n) \equiv \Psi_d(x_0 + m\Delta x, y_0 + n\Delta y)$$
$$u(p,q) \equiv \psi(x_0' + p\Delta x', y_0' + q\Delta y').$$

The integral in Equation 4.6 becomes a finite summation over the discrete functions, yielding the discrete shifted Fresnel transform:

$$U_d(m,n) = \frac{1}{i\lambda d} e^{ikd} e^{i\frac{\pi}{\lambda d}\left(x_m^2 + y_n^2\right)} \sum_{p=0}^{P-1}\sum_{q=0}^{Q-1} u(p,q) e^{i\frac{\pi}{\lambda d}\left(x_p'^2 + y_q'^2\right)} e^{-i\frac{2\pi}{\lambda d}\left(x_p' x_m + y_q' y_n\right)}. \tag{4.9}$$

To produce an expression suitable for fast computation (i.e. FFT or scaled FFT), we must evaluate the terms within the last exponential of Equation 4.9. The last exponential becomes:

$$e^{-i\frac{2\pi}{\lambda d}\left(x_p' x_m + y_q' y_n\right)} = e^{-i\frac{2\pi}{\lambda d}\left[(x_0' + p\Delta x')(x_0 + m\Delta x) + (y_0' + q\Delta y')(y_0 + n\Delta y)\right]}$$
$$= e^{-i\frac{2\pi}{\lambda d}\left(x_p' x_0 + y_q' y_0\right)} e^{-i\frac{2\pi}{\lambda d}\left(x_0' m\Delta x + y_0' n\Delta y\right)} e^{-i\frac{2\pi}{\lambda d}(\Delta x'\Delta x\, pm + \Delta y'\Delta y\, qn)}. \tag{4.10}$$

Using this result in Equation 4.9 produces a discrete shifted Fresnel transform that is suitable for implementation:

$$U_d(m,n) = \frac{1}{i\lambda d} e^{ikd} e^{i\frac{\pi}{\lambda d}\left(x_m^2 + y_n^2\right)} e^{-i\frac{2\pi}{\lambda d}\left(x_0' m\Delta x + y_0' n\Delta y\right)} \times$$
$$\sum_{p=0}^{P-1}\sum_{q=0}^{Q-1}\left[u(p,q) e^{i\frac{\pi}{\lambda d}\left(x_p'^2 + y_q'^2\right)} e^{-i\frac{2\pi}{\lambda d}\left(x_p' x_0 + y_q' y_0\right)} \right] e^{-i\frac{2\pi}{\lambda d}(\Delta x'\Delta x\, pm + \Delta y'\Delta y\, qn)}. \tag{4.11}$$

This fast shifted Fresnel transform multiplies $u(p,q)$ by the exponentials in the summation within the brackets. The computational time required for this operation grows linearly with the size of u. Note that

25

the last exponential in the summation, $\exp\left[-i\frac{2\pi}{\lambda d}\left(\Delta x'\Delta x\,pm+\Delta y'\Delta y\,qn\right)\right]$, is a discrete two-dimensional scaled Fourier transform with scale parameters $\Delta x'\Delta x/\lambda d$ and $\Delta y'\Delta y/\lambda d$ (see Equation 3.6). The fast scaled Fourier transform is computed using the Convolution Theorem (Section 3.2.2) and has the same complexity as the FFT. A final multiplication is done using the complex terms outside of the summation in Equation 4.11.

The major restriction with our fast shifted Fresnel transform is that the resolutions of $U_d(m,n)$ and $u(p,q)$ must be equal, namely that $M = P$ and $N = Q$. This restriction will be overcome in Chapter 5 where we develop a tiling algorithm to handle source and target wavefronts with different resolutions.

Further simplifications of Equation 4.11 are possible by substituting expressions for x'_p and y'_p into the exponentials inside the brackets and removing the factors that are independent of p and q from the summation. We will not carry this out because it does not effect the overall complexity of the algorithm.

A derivation for a fast shifted Fraunhofer transform is also possible and analagous to steps taken for Fresnel diffraction. It will not be discussed here because this research only utilizes the Fresnel diffraction integral.

Chapters 5 and 6 will present two applications of the fast shifted Fresnel transform developed in this research, but first we will see how Fresnel diffraction can be modeled using a fractional Fourier transform.

4.3   Fresnel Diffraction as a Fractional Fourier Transform

It has been shown that Fresnel diffraction is proportional to a fractional Fourier transform [MO93, Loh93, OM93, OBMO94, MZD$^+$95, GMD96]. The following derivation presented is based on [OZK01]. The geometry used is depicted on Figure 4.2, and consists of two spherical surfaces each tangent to a planar surface. For simplification we will consider diffraction in only one dimension. Thus the spherical surfaces are actually circles and the planar surfaces are lines.
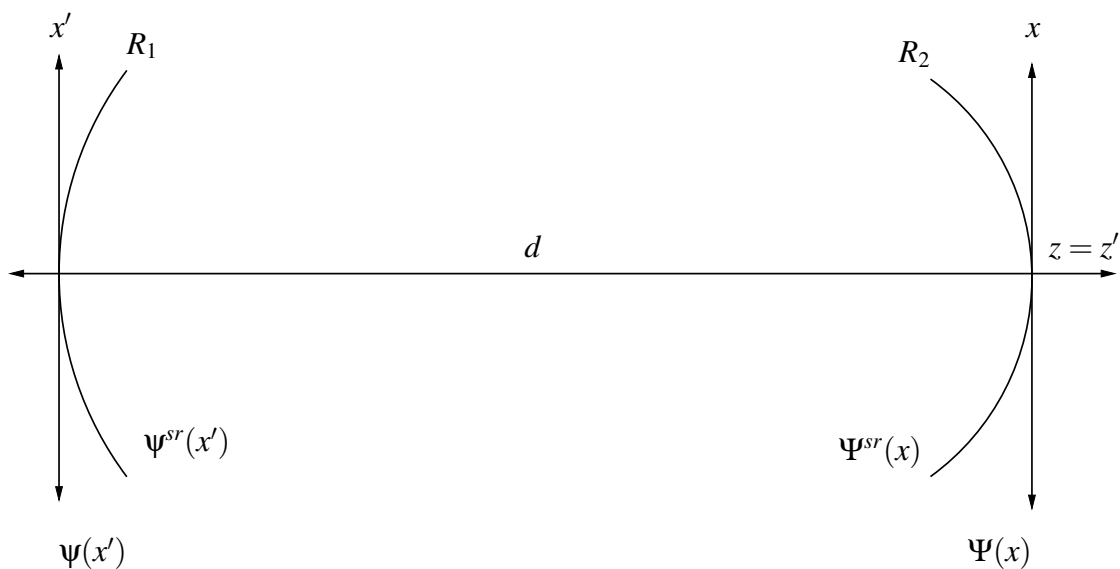

Figure 4.2: Geometry for fractional Fourier modeling of Fresnel diffraction.

$\psi(x')$ represents the wavefront on the first planar surface, $\psi^{sr}(x')$ represents the wavefront on the first spherical surface, $\Psi^{sr}(x)$ represents the wavefront on the second spherical surface, and $\Psi(x)$ represents the wavefront on the second planar surface. These functions are related by:

$$\psi(x') = e^{i\frac{\pi}{\lambda R_1}x'^2}\,\psi^{sr}(x')$$

$$\Psi(x) = e^{i\frac{\pi}{\lambda R_2}x^2}\,\Psi^{sr}(x)\,.$$

(4.12)

The one-dimensional Fresnel diffraction integral, simplified from Equation 4.3, relates the propagated wavefront between the two planar surfaces:

$$\Psi_d(x) = \frac{1}{i\lambda d}\,e^{ikd}\int\limits_{-\infty}^{\infty}\psi(x')\,e^{i\frac{\pi}{\lambda d}(x'-x)^2}\,dx'\,.$$

(4.13)

Substitution of Equation 4.12 into Equation 4.13 allows $\Psi_d^{sr}(x)$ to be expressed as:

$$e^{i\frac{\pi}{\lambda R_2}x^2}\,\Psi_d^{sr}(x) = \frac{1}{i\lambda d}\,e^{ikd}\int\limits_{-\infty}^{\infty}e^{i\frac{\pi}{\lambda R_1}x'^2}\,\psi^{sr}(x')\,e^{i\frac{\pi}{\lambda d}(x'-x)^2}\,dx'$$

$$\Psi_d^{sr}(x) = \frac{1}{i\lambda d}\,e^{ikd}\int\limits_{-\infty}^{\infty}\psi^{sr}(x')\,e^{i\frac{\pi}{\lambda d}(x'-x)^2}\,e^{i\frac{\pi}{\lambda R_1}x'^2}\,e^{-i\frac{\pi}{\lambda R_2}x^2}\,dx'$$

(4.14)

$$\Psi_d^{sr}(x) = \frac{1}{i\lambda d}\,e^{ikd}\int\limits_{-\infty}^{\infty}\psi^{sr}(x')\,e^{i\frac{\pi}{\lambda d}\left[(1+d/R_1)x'^2-2xx'+(1-d/R_2)x^2\right]}\,dx'\,.$$

This expression can be discretized by introducing the dimensionless functions

$$U_d^{sr}(m) \equiv \Psi_d^{sr}(m\Delta x)$$

and

$$u^{sr}(p) \equiv \psi^{sr}(p\Delta x')\,,$$

where m and p are integers and Δx and $\Delta x'$ define the x and x' pitch values, respectively. Equation 4.14 becomes:

$$U_d^{sr}(m) = \frac{1}{i\lambda d}\,e^{ikd}\int\limits_{-\infty}^{\infty}u^{sr}(p)\,e^{i\frac{\pi}{\lambda d}\left[\Delta x'^2(1+d/R_1)p^2-2\Delta x\Delta x'\,mp+\Delta x^2(1-d/R_2)m^2\right]}\,dx'\,,$$

(4.15)

where we see that

$$U_d^{sr}(m) = \frac{1}{i\lambda d}\,e^{ikd}\,\frac{1}{A_\alpha}\,\mathcal{F}^a(u^{sr})$$

if and only if

$$(1+d/R_1)\frac{\Delta x'^2}{\lambda d} = \cot(\alpha)$$

$$(1-d/R_2)\frac{\Delta x^2}{\lambda d} = \cot(\alpha)$$

(4.16)

$$\frac{\Delta x\,\Delta x'}{\lambda d} = \csc(\alpha)\,,$$

27

where $\alpha = a\pi/2$ and A_α is defined as in Equation 3.13. This derivation allows for the fractional Fourier transform to be utilized in arbitrary diffraction systems, which may include lenses as well as distances for free space propagation [OZK01]. For the purpose of this research, we let $R_1 \to \infty$ so that $\psi(x') = \psi^{sr}(x')$ represents a planar wavefront. When computing the Fresnel propagation of a wavefront, the values of $\Delta x'$, λ, and d are given. Thus the values of a, R_2, and Δx are determined from Equation 4.16 as [OZK01]:

$$
\begin{aligned}
\frac{a\pi}{2} &= \alpha = \tan^{-1}\left[\frac{\lambda d}{\Delta x'^2}\right] \\
R_2 &= d\left[1 + \frac{\Delta x'^4}{(\lambda d)^2}\right] \\
\Delta x &= \Delta x'\sqrt{1 + \frac{(\lambda d)^2}{\Delta x'^4}}\,.
\end{aligned}
\tag{4.17}
$$

Noting that $u^{sr}(p) = u(p)$ under the condition that $R_1 \to \infty$, Equation 4.15 becomes

$$
U_d^{sr}(m) = \frac{1}{i\lambda d}\,e^{ikd}\int\limits_{-\infty}^{\infty} u(p)\,e^{i\frac{\pi}{\lambda d}\left[\Delta x'^2 p^2 - 2\Delta x \Delta x' mp + \Delta x^2(1-d/R_2)m^2\right]}\,dx'\,,
\tag{4.18}
$$

and $U_d(m)$ may be expressed as

$$
\begin{aligned}
U_d(m) &= \frac{1}{i\lambda d}\,e^{ikd}\,e^{-i\frac{\pi}{\lambda R_2}\Delta x^2 m^2}\int\limits_{-\infty}^{\infty} u(p)\,e^{i\frac{\pi}{\lambda d}\left[\Delta x'^2 p^2 - 2\Delta x \Delta x' mp + \Delta x^2(1-d/R_2)m^2\right]}\,dx' \\
&= \frac{1}{i\lambda d}\,e^{ikd}\,e^{-i\frac{\pi}{\lambda R_2}\Delta x^2 m^2}\,\frac{1}{A_\alpha}\,\mathcal{F}^a(u)\,,
\end{aligned}
\tag{4.19}
$$

where the value of a is determined by Equation 4.17. In this restrictive case, we see that we are not free to choose the sampling rate of $U_d(m)$. It is for this reason that the fractional Fourier transform is not used to model Fresnel diffraction in this research.

Chapter 5

Rectangular Holographic Tiling

The shifted formulas developed in Chapter 4 will be utilized in a technique we refer to as rectangular holographic tiling, or just simply as tiling. This development is motivated by the fact that the imaging volume requires much less resolution than the hologram. Human acuity is limited to approximately 0.175mm [Luc94], meaning that two points closer than 0.175mm appear as one. Typical holographic resolutions are on the order of a micron. Given a hologram and an imaging volume of roughly the same lateral size, the hologram will require a much higher resolution. Thus fast numerical techniques allowing for the calculation of a diffraction pattern between vastly different resolutions is much needed. This chapter introduces the tiling algorithm developed in this research, several tiling demonstrations, and a discussion of the results.

5.1   Rectangular Holographic Tiling Algorithm

In this research, rectangular tiling, or simply tiling, is defined as the subdivision of a finite rectangular planar surface into smaller, possibly overlapping, rectangular planar surfaces referred to as tiles. These tiles form a set, and together their union covers the original planar surface. Figure 5.1 shows a simple tiling example. This restriction to rectangular tiling is due to the use of the fast algorithms developed in Chapter 4.
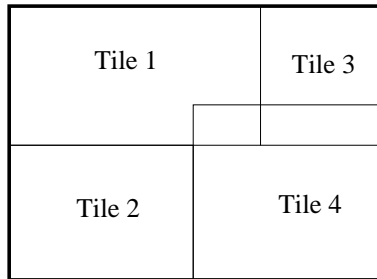

Figure 5.1: Tiling of a finite rectangular plane.

The core of holographic computations uses the propagation and diffraction of light through space, namely between the imaging volume and the hologram. As mentioned in the introduction to this chapter, we will provide for different resolutions for the imaging volume and the hologram. Our tiling construct permits this, in either the construction or reconstruction of holograms. We will often refer to the diffraction between a source plane and a target plane. For hologram construction, the source plane is in the imaging volume and the target plane is the hologram; and vice versa for hologram reconstruction.

We start with a source plane on which the values of a complex wavefront are known, and a target plane where we will calculate the values of the propagated wavefront. Let $\psi(x',y')$ represent the complex-valued wavefront at the source plane, and $\Psi(x,y)$ represent the complex-valued wavefront at the target plane. If ψ and Ψ have the same sampled x and y resolutions, then the computation makes direct use

of the fast discrete shifted Fresnel transform (Equation 4.11). Otherwise, we must consider two cases: when the x and y resolutions of Ψ are not less than that of ψ, and when the x and y resolutions of ψ are not less than that of Ψ. In this research, we will not consider the case where ψ and Ψ have different x and y resolutions, yet neither has a larger resolution in both the x and y directions, an example of which is depicted on Figure 5.2.
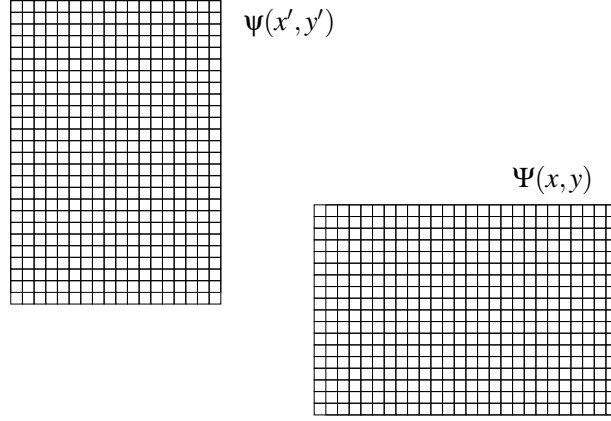

Figure 5.2: Example of holographic tiling case not considered.

When the source wavefront, $\psi(x',y')$, is sampled at a lower resolution than the target wavefront, $\Psi(x,y)$, we create a tiling of Ψ such that each tile has the same resolution as that of ψ, as illustrated in Figure 5.3. Then we sequentially compute the propagated wavefront of ψ for each tile. Since each tile has the same resolution as ψ, the fast shifted Fresnel transform may be used (Equation 4.11).
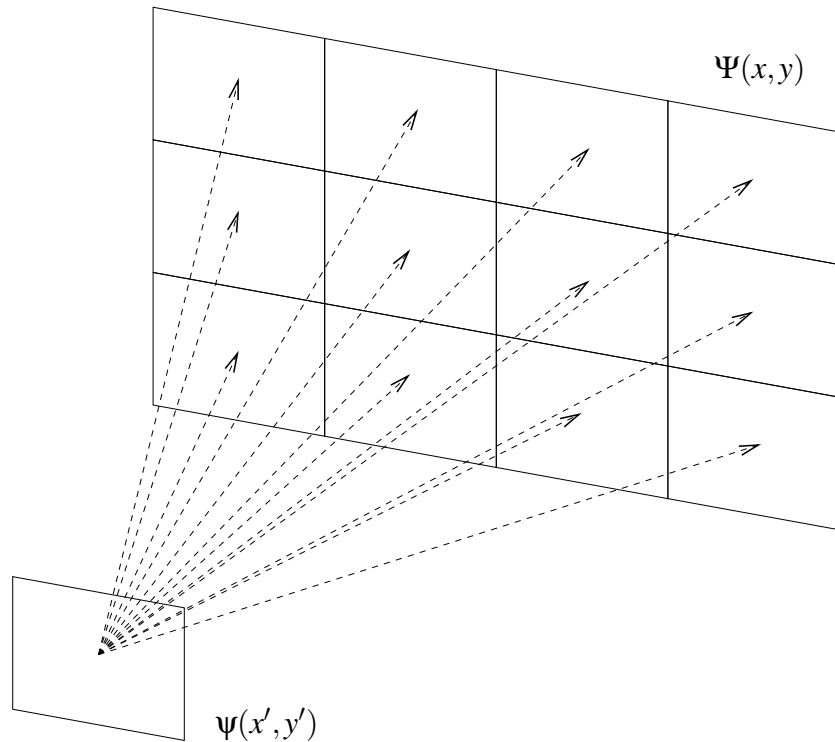

Figure 5.3: Lower resolution source wavefront for holographic tiling.

When the target wavefront, $\Psi(x, y)$, is sampled at a lower resolution than the source wavefront, $\psi(x', y')$, we use the superposition principle of light. Create a tiling of ψ such that each tile has the same resolution as that of Ψ, as illustrated in Figure 5.4. The value of Ψ is equal to a linear superposition of the propagated wavefront of each tile of ψ. Thus the propagated wavefront from each tile at the location of Ψ is summed.
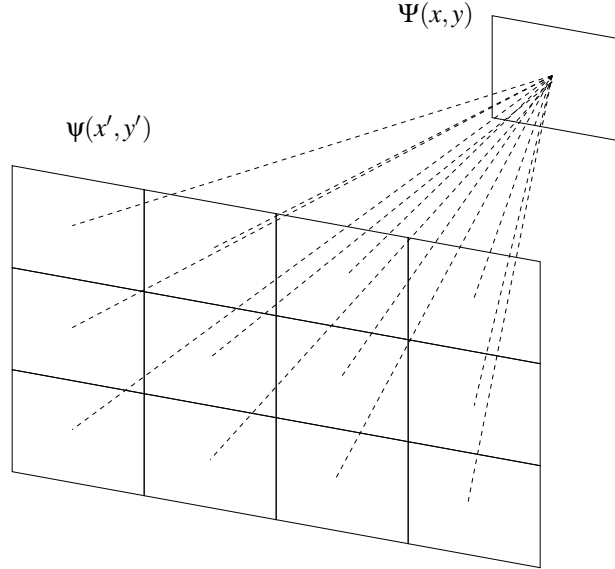

Figure 5.4: Lower resolution target wavefront for holographic tiling.

5.2 Holographic Tiling Demonstrations

In this section, several applications of our holographic tiling algorithm are demonstrated. Each is implemented using Java, executed on an Intel Dothan PC under Linux, and represents a prototype implementation.

5.2.1 Using Low Resolution Source Images

In traditional Fourier-based hologram calculations, the source and target wavefronts must have the same resolution. This first demonstration will use the tiling algorithm to compute a high resolution hologram from relatively low resolution source images. It will show how the computational time is reduced by the use of smaller resolution source images, when the use of small resolution images is permissable.

The image in Figure 5.5 was used as the source image, having the following resolutions: 512×512, 256×256, 128×128, 64×64, and 32×32. For each resolution, the tiling algorithm was applied to construct a 1024×1024 resolution hologram having an x and y pitch of 8 microns. In each case, the source image was 6mm by 6mm and located 50cm from the hologram. The spatial pitch of the source image for each trial varied since the resolution was different but the total size was constant. Figure 5.6 displays the holograms computed and the reconstructions for each resolution. A good reconstruction was accomplished for each resolution, where image degradation seen in the lower resolutions is due to the

lower resolution of the source image. The reconstruction of each hologram was also carried out using the tiling algorithm to reconstruct the hologram at the same resolution as each source image. During the reconstruction phase, the hologram became the source plane and the imaging area became the target plane, thus utilizing the superposition principle.

Figure 5.5: Source image for low resolution tiling demonstration.

Table 5.1 displays the results of this demonstration. As expected, using lower resolution source images takes less computational time, although not as significant as one might think. To better understand

Table 5.1: Low resolution source image tiling demonstration results.

Source Resolution	Hologram Computation Time (seconds)
512×512	30.86
256×256	26.39
128×128	11.03
64×64	9.20
32×32	8.80

the decrease in computational time, we shall determine the complexity of the tiling algorithm. Let the hologram have a resolution of $N \times N$, and let the source image have a resolution of $M \times M$, such that $2^K M = N$ for some non-negative integer K. The number of tiles generated to cover the target plane will be $(2^K)^2$ and the complexity to transform each tile will be $M \log M$. Therefore the total complexity is:

$$(2^K)^2 \, M^2 \log M = (2^K)^2 \left(\frac{N}{2^K} \right)^2 \log \left(\frac{N}{2^K} \right) = N^2 (\log N - K).$$

We see that the computational costs decrease linearly with K.

A possible future area of research for this tiling algorithm would be to automatically determine, given an input souce image consisting of sparse data, if it is more computationally efficient to divide the image into
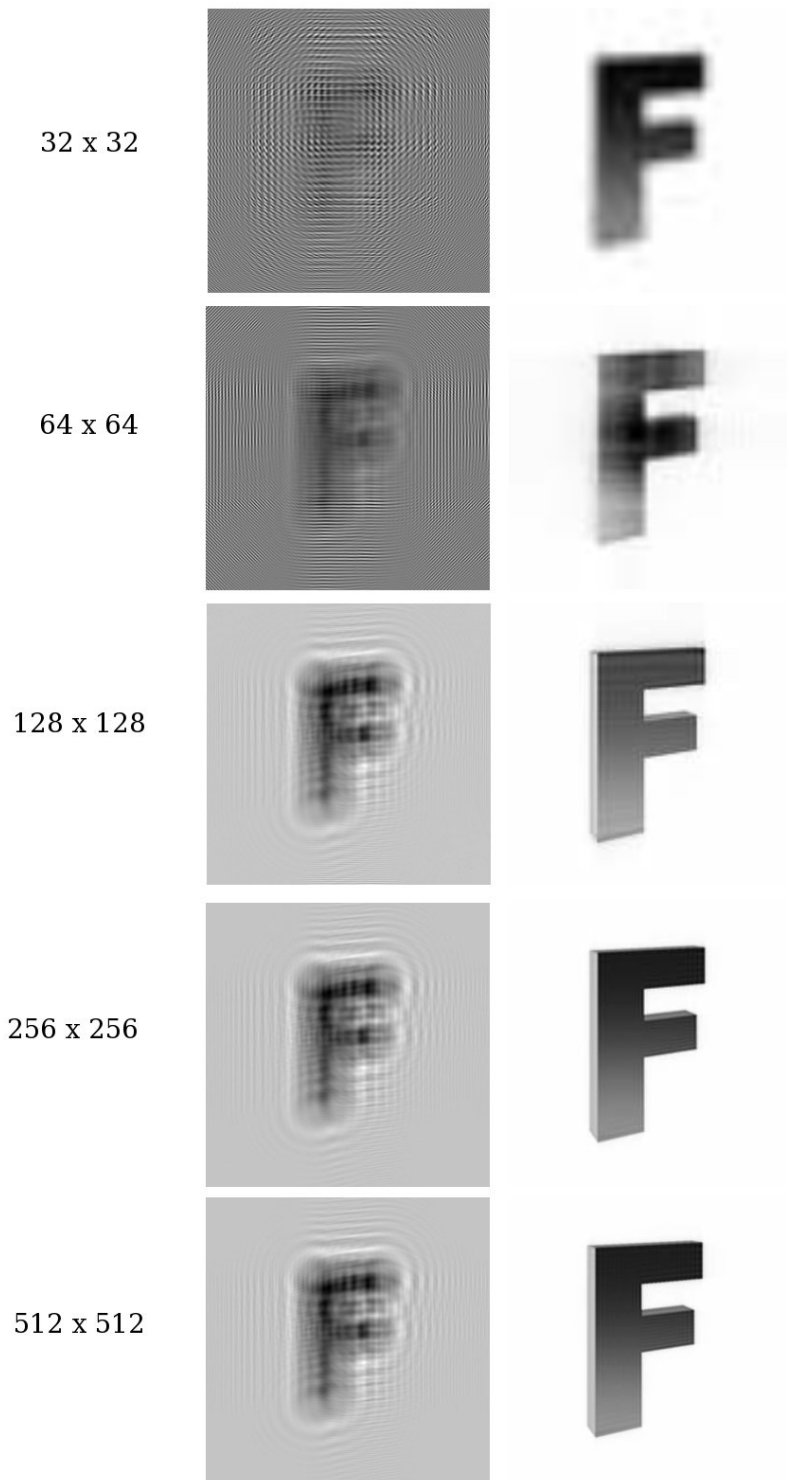
Figure 5.6: Reconstructed images from low resolution tiling demonstration. The images in the left column are the computed holograms, and the images in the right colum are the reconstructed images.

smaller resolution blocks and transform each separately. This may prove more effective when the input source image consists of large areas of "blank" data, which do not contribute to the final holographic fringe pattern.

5.2.2 Tiling Images with Varying Depths

This demonstration will illustrate the effectiveness of the shifted diffraction formulas derived in Chapter 4 while using the tiling algorithm. Three source images were used, each having a resolution of 256×256 and a size of 4mm by 4mm. Figure 5.7 shows the three images used. Image A was located at $(-2\text{mm}, 2\text{mm}, 50\text{cm})$, Image B was located at $(0\text{mm}, 0\text{mm}, 52\text{cm})$, and Image C was located at $(2\text{mm}, -2\text{mm}, 54\text{cm})$. A single hologram was calculated having a resolution of 1024×1024 with an x and y pitch of 8 microns. The hologram computed is illustrated on Figure 5.8.
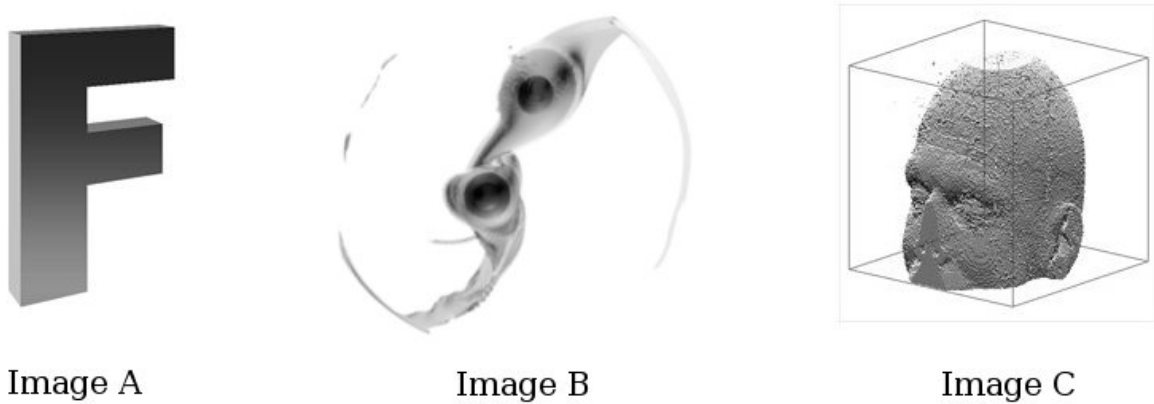

Image A   Image B   Image C

Figure 5.7: Source images having various depths for tiling demonstration.

The reconstruction was carried out for the depth of each source image, as shown on Figures 5.9, 5.10, and 5.11. In each reconstruction, the proper image is in focus, located in the proper location and orientation, and the other two images are seen as out of focus.
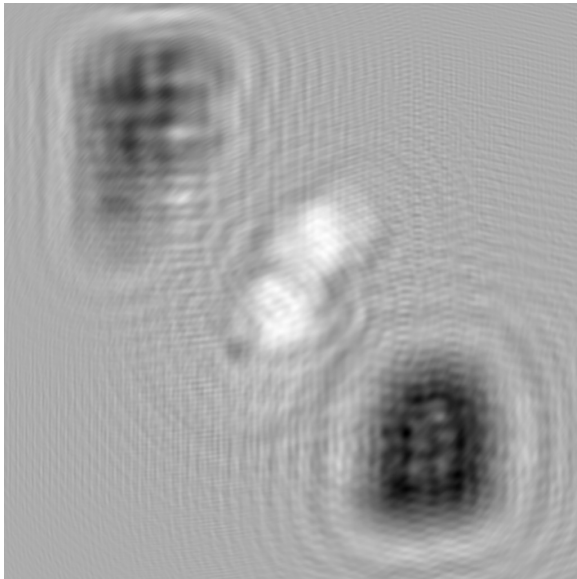

Figure 5.8: Hologram computed from three source images using the tiling algorithm.

Figure 5.9: Reconstruction of Image A for a tiling demonstration.

Figure 5.10: Reconstruction of Image B for a tiling demonstration.

Figure 5.11: Reconstruction of Image C for a tiling demonstration.

5.2.3 Real Image Reconstruction

As discussed in Section 2.3.3, real image reconstruction involves illuminating only a small portion of the hologram and thus producing a single perspective of the object. Each region of the hologram is oriented at a certain angle to the object, and thus only stores information from that particular perspective of the object. Tajahuerce showed that computational reconstruction from a small rectangular portion of a hologram indeed produces the proper perspective of the original object [TJ00]. Our tiling algorithm will now be applied to simulate the real image reconstruction of a hologram.

Our approach is to tile the hologram into non-overlapping regions and reconstruct the object using each tile independently. No superpositioning is carried out. The result is that each tile of the hologram reconstructs the proper perspective. The object from which the hologram is computed is a cube with a pointed top. Vertices are generated to lie along the edges of the object, and an interference-based approach is taken to compute the hologram. Figure 5.12 shows a 256×256 portion of the 1024×1024 hologram computed from the object. Figures 5.13, 5.14, and 5.15 illustrate real image reconstructions using tile resolutions of 512×512, 256×256, and 128×128, respectively. The object is approximately 1mm wide and located 3cm from the hologram. The object is tilted prior to hologram construction so that one corner of the cube is closest to the hologram. The x and y pitch of the hologram is chosen to be 14 microns. These parameters are chosen to produce a good illustration of the various perspectives of the object.

Note that each tile reconstructs the proper perspective of the object. Also, the reconstructions are carried out for a plane through the cube. Therefore points in front of or behind this plane are blurred. This effect is much more pronounced in the higher resolution tiles.

In principle, this same process will work in reverse. Given a three-dimensional object, we would begin by generating a discrete number of two-dimensional perspectives. These perspectives would each represent a tile and would be independently transformed to produce the final hologram. This technique will not be
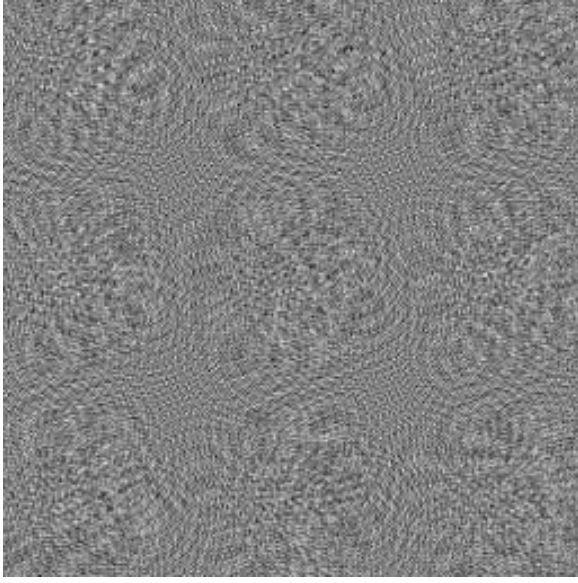
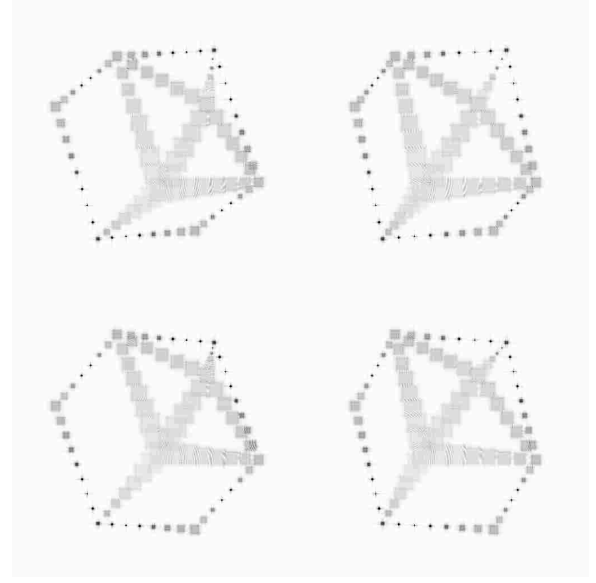Figure 5.12: Hologram produced to illustrate real image reconstruction.

Figure 5.13: Reconstructed real images using a tile size of 512×512.
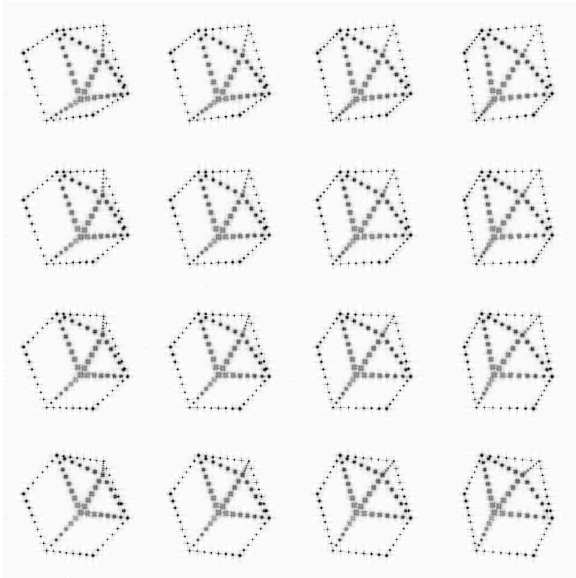

Figure 5.14: Reconstructed real images using a tile size of 256×256.
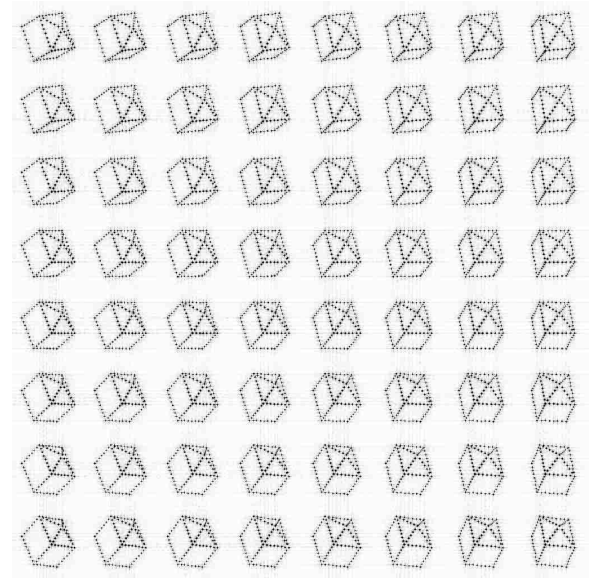

Figure 5.15: Reconstructed real images using a tile size of 128×128.

pursued here, and the interested reader may refer to the following sources for similar approaches: [AR03, NOY+97, SIY04].

5.3 Holographic Tiling Discussion

Holographic computations are easy to program in parallel, since the computation of any region of the hologram can be done independently from any other region. Our tiling algorithm preserves this attribute

since each tile can be computed independently from the others. This same property can be used when we wish to compute an extremely large hologram that cannot be entirely stored in memory; each tile of the hologram can be computed separately and stored for later use. Also, a process similar to scan line imaging of two-dimensional displays may be needed for real-time holographic devices. It may be advantageous to compute tiles of the hologram sequentially, constantly sending each resulting fringe pattern to a device for display. The following chapter will make further use of the tiling algorithm in what we call partitioned holographic computation.

Chapter 6

Partitioned Holographic Computation

In two-dimensional rendering of three-dimensional data, it is common to use a bounded volume heirarchy [AMH02]. Each object in a scene is contained within a simpler geometric solid, such as a cube, sphere, or cylinder. These may also be grouped and contained within larger solids. During rendering, the determination of visibility, occlusion, or intersection of objects are performed on the geometrically simpler bounding volumes before the objects themselves are considered. It is this general procedure that inspired the development of partitioned holographic computation in this research.

Partitioned holographic computation divides a scene into subsets of objects, where an intermediate hologram is computed for each subset using only the objects in that subset. Then the final hologram is computed using the set of intermediate holograms. We assume each intermediate hologram to be independent, i.e. changes to one subset of objects does not effect the computation of the other intermediate holograms. This is a viable approach to computing the final hologram, as long as the conditions that follow are met.

We require that if occlusion is to be properly modeled, then objects in one subset must not occlude objects in another subset. This would violate our assumption that the intermediate holograms are independent. Since Fresnel diffraction is used in this research, a minimum distance of propagation is required (restated from Equation 4.4):

$$d \geq \frac{\Delta x^2 \, N + \Delta y^2 \, M}{\lambda},$$

(6.1)

where Δx and Δy are the x and y pitch values, respectively, and N and M are the x and y resolutions, respectively. This condition must hold for the distance from the objects in each subset to its corresponding intermediate hologram, as well as the distance from each intermediate hologram to the final hologram. This means that the lateral dimension of the intermediate hologram must be of sufficient size to properly represent the propagating wavefront between each object and the hologram.

The long term goal of the partitioned hologram formulation is to provide a more dynamic framework for computing holograms that will aid the development of real-time holographic displays. This approach represents a hybrid, where we have made no assumptions on the exact form of the objects. For example, one subset may contain a point-based object, its intermediate hologram would be computed using an interference-based, ray-tracing approach. Another subset may contain a collection of planar objects, computation of its intermediate hologram would use the shifted Fresnel transform.

This chapter introduces the partitioned holographic computation algorithm developed in this research, two partitioning demonstrations, and a discussion of the results.

6.1 Partitioned Holography Algorithm

First consider the propagating wavefront emmitted from an object. The value of the wavefront at any distance can be seen as a different representation of the same signal. The holographic process relies on

the ability to capture this wavefront at the plane of the hologram, and then to propagate this wavefront to reconstruct the original object. Our partitioned holography algorithm uses this property by introducing intermediate holograms to reduce the computational time.

We are given a set of objects from which a hologram is to be computed. This hologram will be referred to as the final hologram. This set of objects is partitioned so that each object belongs to exactly one subset. For each subset, we define an intermediate hologram. Each intermediate hologram is computed using only the objects contained in that subset. The final hologram is then computed using only the intermediate holograms, calculated by the linear superposition of the propagated wavefronts from each intermediate hologram. Since the shifted Fresnel diffraction techniques developed in Section 4.2 require propagation between parallel planes, this research only considers intermediate holograms that are parallel to the final hologram.

As mentioned previously, the location of each intermediate hologram must satisfy the minimum Fresnel distance (Equation 6.1) to the objects in question and to the final hologram. If the objects are located a sufficient distance from the final hologram, then the intermediate hologram may be located as depicted on Figure 6.1. If this is not the case, then the intermediate hologram may be located behind the hologram as depicted on Figure 6.2.
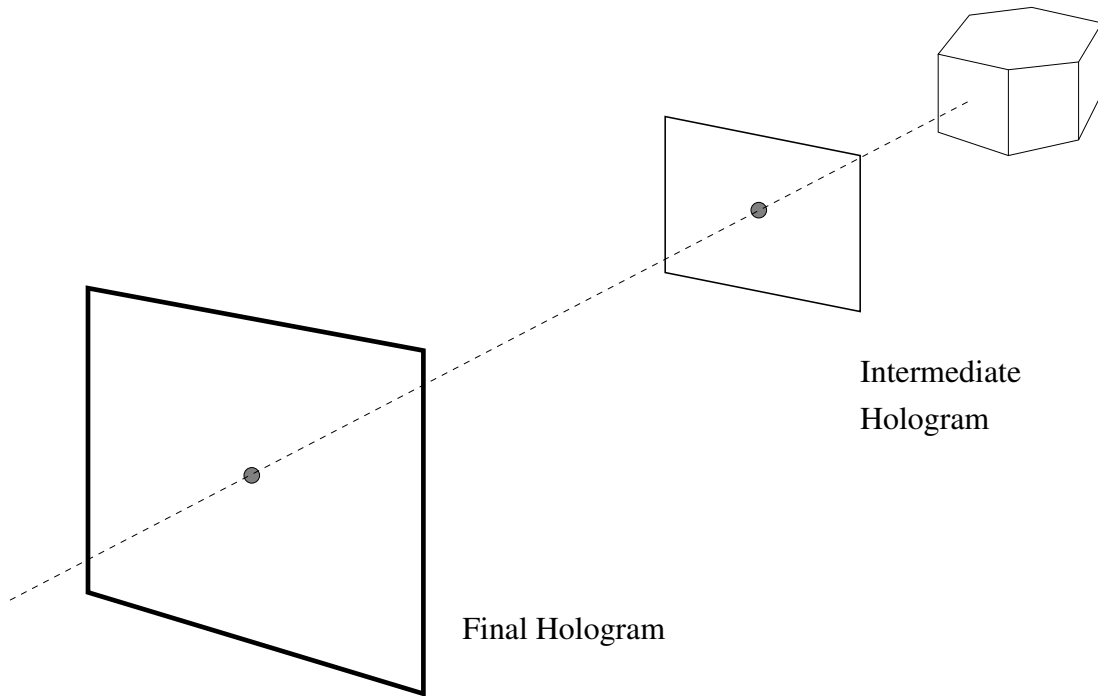

Figure 6.1: Diagram for partitioned holographic computation.

Locating the intermediate hologram and the object on opposite sides of the final hologram may seem counter intuitive. But the intermediate hologram is simply acting as a "placeholder", storing a wavefront which represents the propagating wavefront from the object at some plane. Assume the object is located a distance d from the final hologram, and the intermediate hologram is located a distance d' behind the final hologram (see Figure 6.2). The object wavefront is first propagated a distance of $d + d'$ to the location of the intermediate hologram, then this wavefront is propagated back a distance $-d'$ to the location of the final hologram.
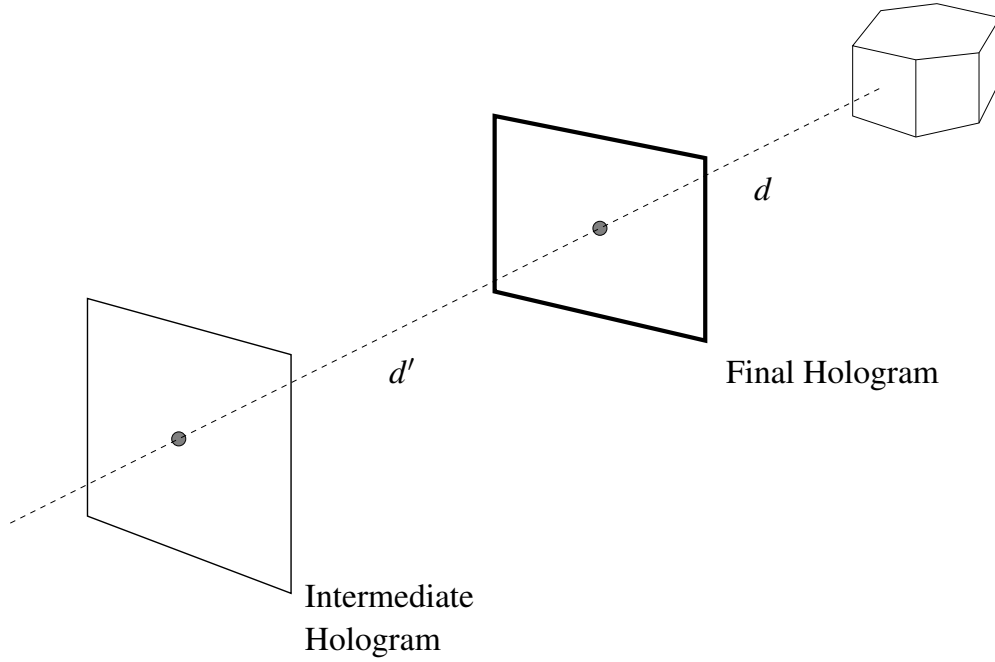
Figure 6.2: Alternate diagram for partitioned holographic computation.

6.2 Partitioned Holography Demonstrations

In this section, several applications of our partitioned holographic computation are demonstrated. The main advantage of these partitioned holographic computations are that the intermediate holograms require less resolution than the final hologram. Each is implemented using Java, executed on an Intel 1.7GHz Dothan PC with Linux using 1GB of RAM, and represents a prototype implementation.

6.2.1 Partitioned Hologram of Planar Objects

This example provides a "proof of principle" demonstration of our partitioned holographic computation. The same source images depicted on Figure 5.7 are used, each having a resolution of 256×256 and a size of 4mm by 4mm. Image A is located at $(-2\text{mm}, 2\text{mm}, 80\text{cm})$, Image B is located at $(0\text{mm}, 0\text{mm}, 82\text{cm})$, and Image C is located at $(2\text{mm}, -2\text{mm}, 84\text{cm})$. A single intermediate hologram is used having a resolution of 512×512, a size of 1.7cm by 1.7cm, and is centered at $(0, 0, 40\text{cm})$. The intermediate hologram is calculated from these three source images, and the result is illustrated on Figure 6.3. The final hologram, calculated solely from the intermediate hologram, is shown on Figure 6.4, and has a resolution of 1024×1024 with an x and y pitch of 17 microns. Figures 6.5, 6.6, and 6.7 depict the reconstructions carried out for each depth of the source images.

The time required to compute the intermediate hologram was 17.8 seconds, and the time to compute the final hologram from the intermediate was 33.0 seconds; for a total computational time of 50.8 seconds. For comparison, computation of the final hologram directly from the three source images took 84 seconds. Thus for this simple example, using a single intermediate hologram proves viable in hologram construction and decreases the total computational time needed by approximately 40%.
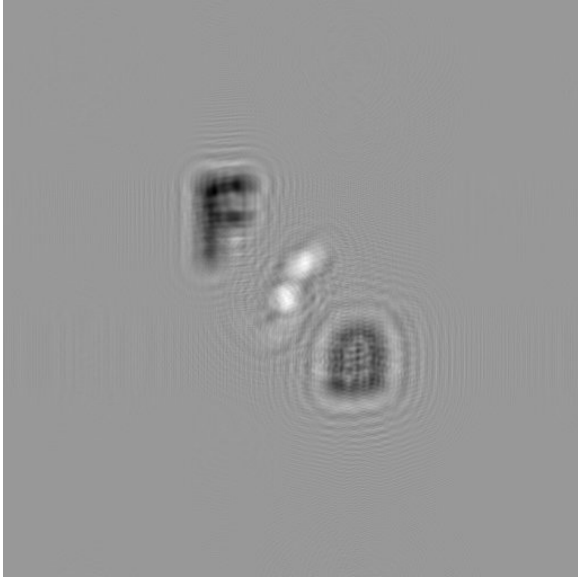
Figure 6.3: Intermediate hologram computed for planar partitioning demonstration.
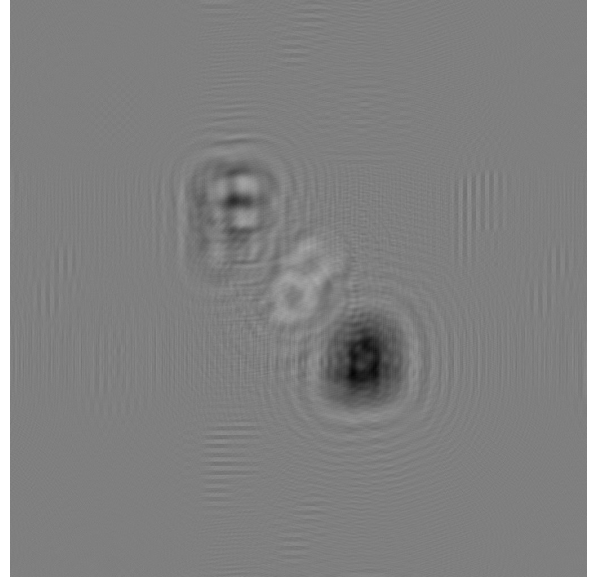

Figure 6.4: Final hologram computed for planar partitioning demonstration.

Figure 6.5: Reconstruction of Image A for planar partitioning demonstration.

Figure 6.6: Reconstruction of Image B for planar partitioning demonstration.

6.2.2 Partitioned Hologram of a Point-Based Object

Figure 6.8 shows a rendering of four isosurfaces from a binary star simulation. Only the vertices from the outermost isosurface are used in this demonstration for the construction of a hologram. The surface consists of 21670 vertices, and an interference-based, ray-tracing algorithm is implemented to calculate a single intermediate hologram. The surface is rotated before the hologram computation, thus reconstructions show a different view than shown on Figure 6.8. Three configurations are used to produce holograms of these data, each resulting in a need for different placement of the intermediate hologram.
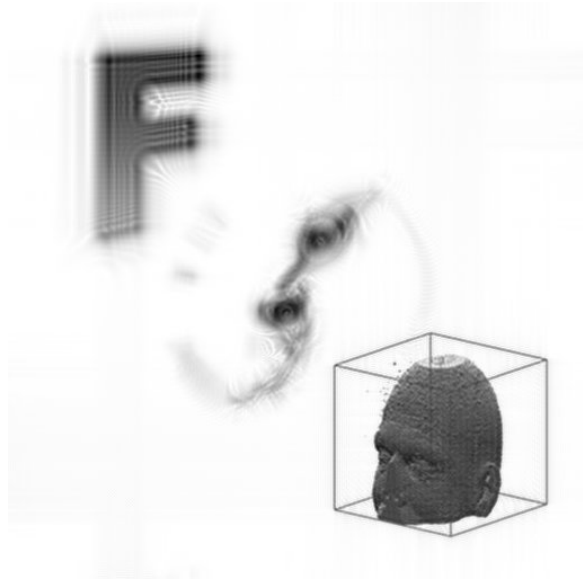
41

Figure 6.7: Reconstruction of Image C for planar partitioning demonstration.
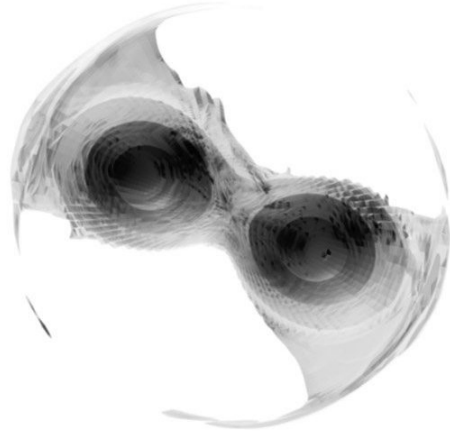

Figure 6.8: A rendering of isosurfaces from a binary star simulation.

In the first configuration considered, the final hologram has a resolution of 1024×1024 and a pitch of 4 microns. The isosurface is centered at (0, 0, 10cm) and scaled to have an x dimension of 3mm. The intermediate hologram is centered at (0, 0, 5cm), has a resolution of 512×512, and set to have the same size as the final hologram, roughly 4mm by 4mm. Note that the intermediate hologram is located between the object and the final hologram. Figure 6.9 illustrates the intermediate hologram and Figure 6.10 illustrates the final hologram. The reconstruction of this hologram is shown on Figure 6.11. For comparison, the final hologram is also computed directly from the source object, and its reconstruction is shown on Figure 6.12.
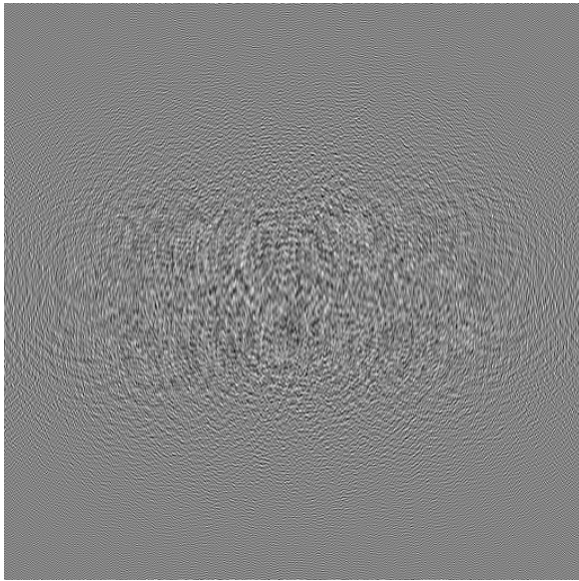

Figure 6.9: Intermediate hologram for partitioned hologram demonstration using the first configuration.
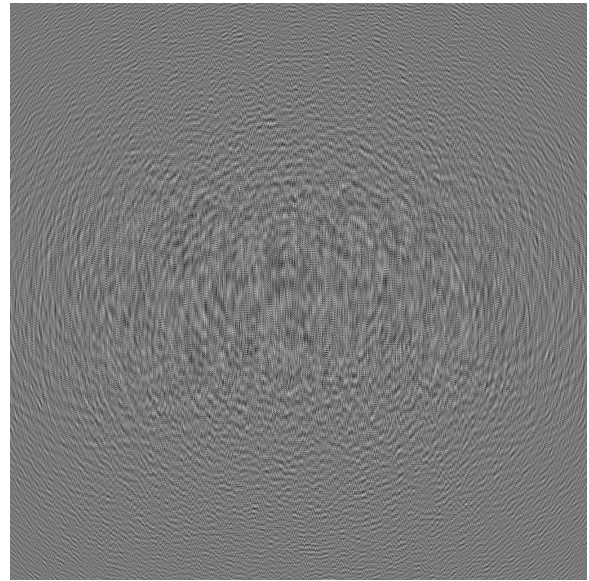

Figure 6.10: Final hologram for partitioned hologram demonstration using the first configuration.
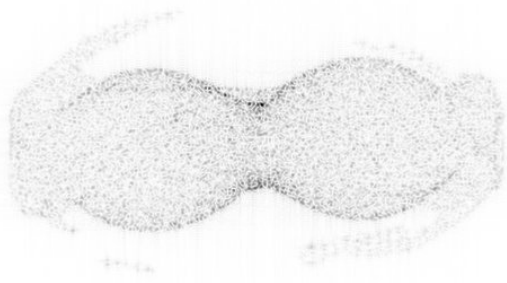
Figure 6.11: Reconstruction from partitioned hologram demonstration using the first configuration.
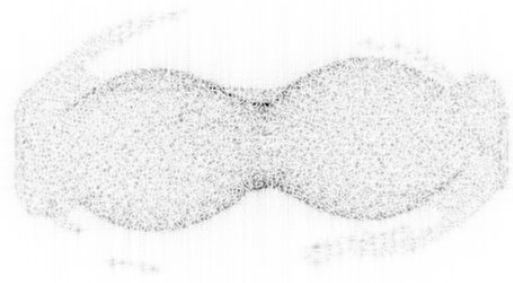

Figure 6.12: Reconstruction from direct computation using the first configuration.

In the second configuration considered, the final hologram has a resolution of 1024×1024 and a pitch of 17 microns. The isosurface is centered at (4cm, 2cm, 122cm) and scaled to have an x dimension of 3.5cm. The intermediate hologram is centered at (-8.5cm, -4.3cm, -260cm) and set to have the same x dimension as the original object. Note that the final hologram is located between the object and the intermediate hologram, and that a line may be drawn passing through the centers of each. Figure 6.13 illustrates the intermediate hologram and Figure 6.14 illustrates the final hologram. The reconstruction of this hologram is shown on Figure 6.15. For comparison, the final hologram is also computed directly from the source object, and its reconstruction is shown on Figure 6.16.

Figure 6.13: Intermediate hologram for partitioned hologram demonstration using the second configuration.
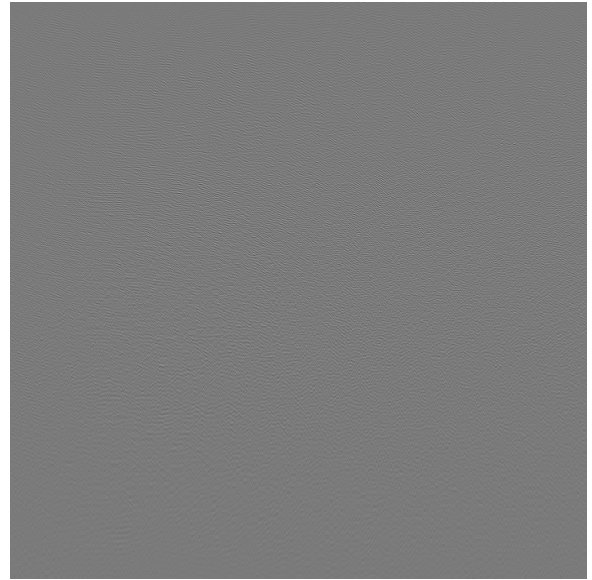
Figure 6.14: Final hologram for partitioned hologram demonstration using the second configuration.

The computational times for these two configurations are approximately the same since each setup uses the same resolutions, the times given are averages. The time needed for direct calculation of the final hologram is 4200 seconds. The time needed to compute the intermediate hologram is 940 seconds and to compute the final hologram is 35 seconds; resulting in a total computational time for the partitioned approach of 975 seconds. We see a reduction in computational time using the partitioned approach of approximately 75%.
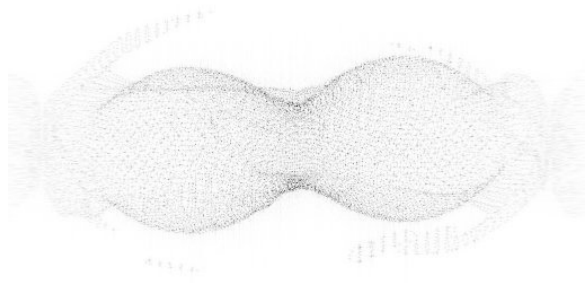
Figure 6.15: Reconstruction from partitioned hologram demonstration using the second configuration.
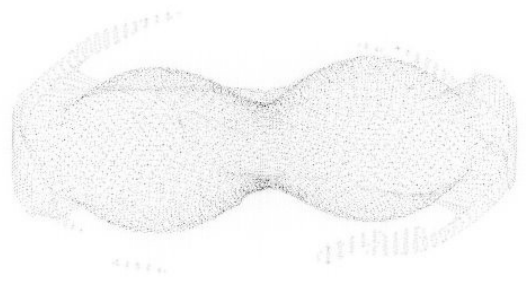
Figure 6.16: Reconstruction from direct computation using the second configuration.

Seeking to reduce the required computational time, the third configuration modifies these parameters so that the intermediate hologram has a resolution of 256×256. To respect the required minimum Fresnel distance, we center the intermediate hologram at $(-12\text{cm}, -6\text{cm}, -366\text{cm})$, maintaining the same x dimension as the object. Figure 6.17 illustrates the intermediate hologram and Figure 6.18 illustrates the final hologram. The reconstruction of this hologram is shown on Figure 6.19. The time to compute the intermediate hologram decreased to 240 seconds and the final hologram to 28 seconds, resulting in a total computational time of 268 seconds. Note that the time to compute the intermediate hologram decreases by a larger factor, since its computation uses an interference-based approach with complexity $O(N^4)$.
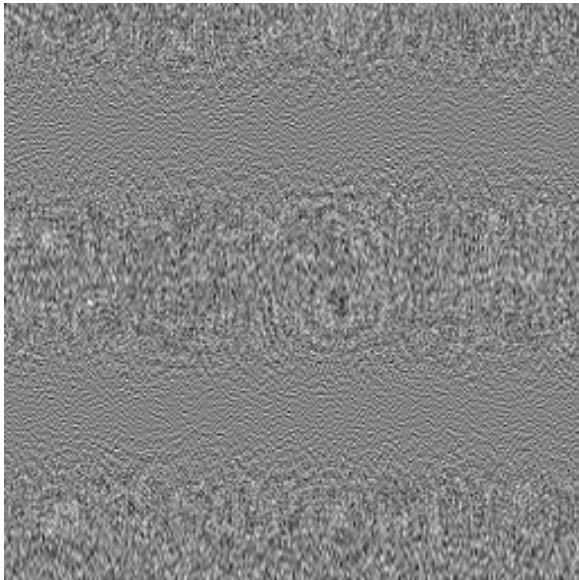
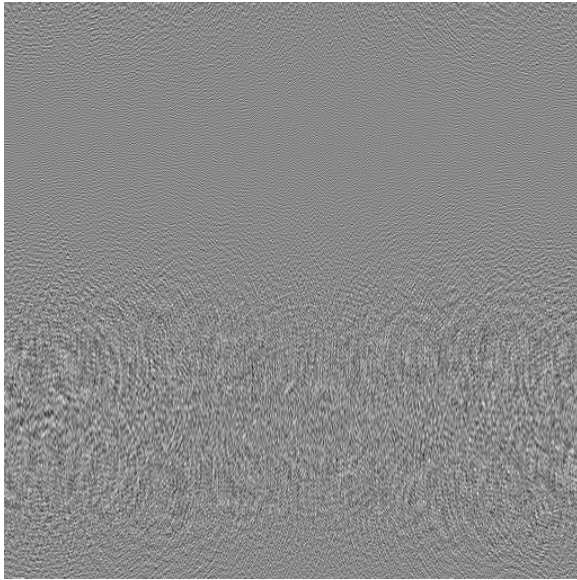

Figure 6.17: Intermediate hologram for partitioned hologram demonstration using the third configuration.
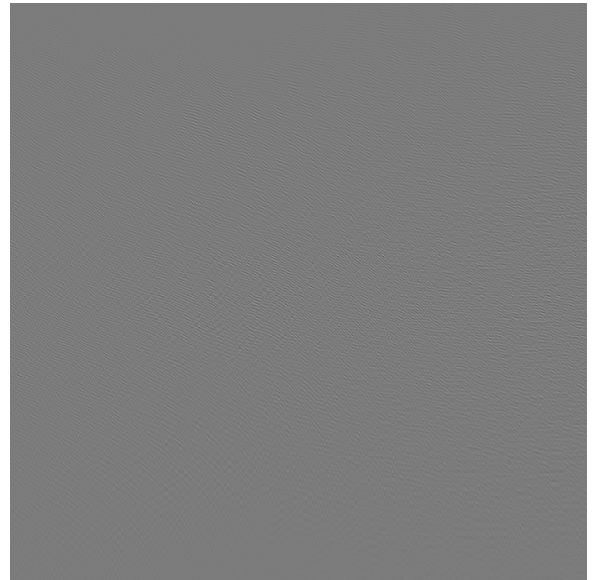
Figure 6.18: Final hologram for partitioned hologram demonstration using the third configuration.

6.3 Partitioned Holography Discussion

Our demonstrations in this chapter utilized a single intermediate hologram for simplicity to show "proof of principle". Future research may focus on the development of a real-time framework which would
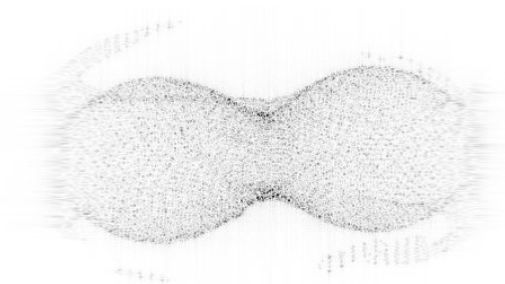
Figure 6.19: Reconstruction from partitioned hologram demonstration using the third configuration.

allow arbitrarily nested objects and intermediate holograms represented by a tree-like structure. The updating of a single object would only require recomputing the intermediate holograms which it effects. Also, the quantity, location, and resolutions of the set of intermediate holograms would ideally be dynamic.

The choice of using rectangular intermediate holograms parallel to the final hologram was made so that our shifted Fresnel transform could be utilized. The future development of fast algorithms for non-planar holograms would allow investigation into intermediate holograms on other surfaces, such as spheres or cylinders. For instance, assume we calculate the wavefront generated by an object at the surface of a sphere which encloses that object. The contribution to the final hologram would involve propagating the wavefront on the portion of the sphere which faces the hologram. Thus translations and rotations of the object would not require recalculating the spherical wavefront, but merely propagating a different portion of the spherical wavefront.

Chapter 7

Conclusion

7.1 Research Overview

The discrete shifted Fresnel transform has been derived in this research (Section 4.2) to overcome disadvantages of traditional Fourier techniques, namely that the sampling rates are fixed and the sampled points of the hologram and object must both be centered along the axis of propagation (Section 4.1). This shifted Fresnel transform permitted computation of Fresnel diffraction between two parallel planar surfaces. An efficient implementation, the fast shifted Fresnel transform, was developed that utilized a fast scaled Fourier transform for computation.

To compute Fresnel diffraction between planar surfaces with different resolutions, we developed a tiling algorithm which utilized the shifted Fresnel transform. The cases where the source plane or target plane had a higher resolution were both handled well, and demonstrations of the technique for constructing and reconstructing holograms were successful. The use of low reslution source images to compute a relatively high resolution hologram was demonstrated, and it was seen that computational times decrease linearly with the size of the source image. Simulated real image reconstruction was carried out using the tiling algorithm, generating a set of 2D perspective images representing the three-dimensional source object.

The partitioned holographic computation technique was introduced as an alternate hologram construction method. In this approach, the imaging volume was divided into subsets of objects. An intermediate hologram was computed from each subset, and the final hologram was computed using only the intermediate holograms. Our demonstrations showed examples using a single intermediate hologram, and either planar objects or point-based objects. Computation of an intermediate hologram from planar objects could use of our shifted Fresnel transform, but an interference-based, ray-tracing calculation had to be performed with the point-based object. The largest decrease in computational time was seen when using a point-based object. We were able to use a relatively low resolution intermediate hologram for use in the interference-based calculations, then proceed in calculating the final hologram using the fast shifted Fresnel transform.

7.2 Future Directions of Research

As previously mentioned, the fast discrete shifted Fresnel transform developed in this research has complexity $O(N \log N)$ for one-dimensional applications and complexity $O(N^2 \log N)$ for two-dimensional applications. This is more efficient than a "brute-force" approach that results in complexity of $O(N^2)$ or $O(N^4)$, yet may still be improved. Fast calculation of our shifted Fresnel transform relied on the computation of a scaled Fourier transform, which in turn relied on, in this research, applying the Convolution Theorem (Equations 3.10 and 3.11) and computing three Fourier transforms. Our computational time could be reduced if an alternative method of computing a scaled Fourier transform were found with fewer than three executions of the FFT.

The minimum allowable propagation distance is imposed when using the Fresnel diffraction formula that stems from the assumptions made in its derivation (Section 4.1). For distances less than the minimum Fresnel distance, the angular spectrum propagation method has been utilized [DNFP+05, SJ94, GFDN+01]. The generalization of angular spectrum propagation to arbitrarily shifted apertures may prove useful in further developments of the partitioned hologram computation method. One drawback is that two computational methods are needed; one for distances less than the minimum Fresnel distance, and one for greater. The fractional Fourier transform, on the other hand, can theoretically describe the propagation of light for any distance from 0 to ∞ [GMD96, PF94, OZK01]. Further research into using this transform in the context of this research may also prove useful in our partitioned hologram computation.

In the development of the fast shifted Fresnel transform, we assumed that the source and target wavefronts were defined on parallel planes. It would be beneficial to extend the formulations in this research to arbitrarily tilted finite rectangular planes as well as other geometric primitives, e.g. cylinders or spheres. There has been research in the diffraction from tilted planes, and the interested reader may see [EO04, MSW02, MK03, LF88, DNFP+05].

Throughout this research we used rectangular grids exclusively. The advantages of using hexagonal grids in 2D image processing have been extensively researched [Per96, PA99, VDVBU+04], yet their use for holographic purposes has not yet been pursued. Advantages of hexagonal grids compared to rectangular grids are that they have a lower sampling density, are more tightly packed, and have three axes of symmetry (60°, 120°, and 180°) as opposed to two (90° and 180°) for rectangular grids. The major disadvantage of hexagonal grids for holographic displays may be in a manufacturing step. We have done some preliminary investigations which show that our tiling algorithm can be used in calculating the Fresnel diffraction of a hexagonal grid by observing that a hexagonal grid can be decomposed into two overlapped, offset rectangular grids. Figure 7.1 illustrates a hexagonal grid showing two overlapped rectangular grids, one made from the filled vertices and the other from the unfilled vertices. A more efficient approach than the tiling algorithm might be obtained by adapting our formulations with the use of existing hexagonal FFT methods [WP84, Ehr93, Gri02, ZR00].
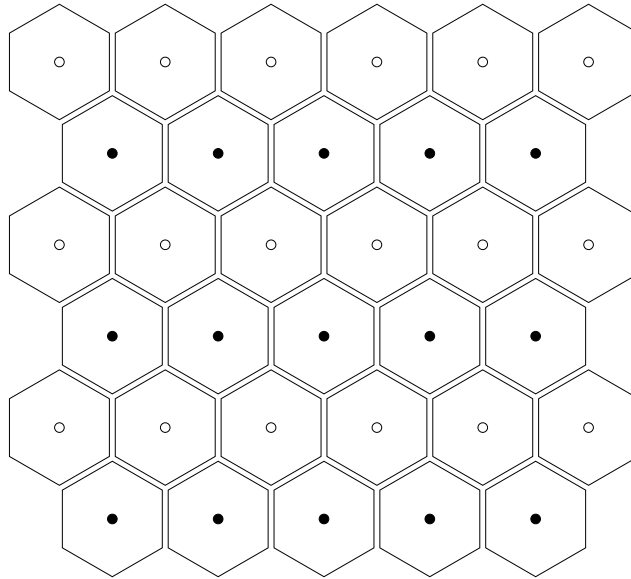
Figure 7.1: Hexagonal grid showing decomposition into two rectangular grids.

7.3   Concluding Remarks

We believe that eventually an ideal method for holographic computations will be developed which capitalizes on the distinct advantages of the interference-based, Fourier-based, and diffraction-specific approaches, while carefully weighing the disadvantages of each. It is our contention that our fast implementation of shifted Fresnel diffraction, the tiling construct, and partitioned hologram computation represent a step in this direction.

Bibliography

[AMH02]   Tomas Akenine-Moller and Eric Haines. *Real-Time Rendering*. AK Peters, Ltd., 2002.

[AR03]   David Abookasis and Joseph Rosen.   Computer-generated holograms of three-dimensional objects synthesized from their multiple angular viewpoints. *Journal of the Optical Society of America A*, 20(8):1537–1545, August 2003.

[BH93]   Michael Barnsley and Lyman Hurd. *Fractal Image Compression*. AK Peters, Ltd., 1993.

[BJF70]   Frederick W. Byron Jr. and Robert W. Fuller. *Mathematics of Classical and Quantum Physics*. Dover Publications, Inc., 1970.

[Boo01]   Ronald G. Boothe. *Perception of the Visual Environment*. Springer, November 2001.

[BS91]   David H. Bailey and Paul N. Swarztrauber. The fractional fourier transform and applications. *SIAM Review*, 33:389–404, 1991.

[CKLL05]   K. Choi, J. Kim, Y. Lim, and B. Lee. Full parallax viewing-angle enhanced computer-generated holographic 3d display system using integral lens array. *Optics Express*, 13:10494–10502, 2005.

[CO96]   Bradley W. Carroll and Dale A. Ostlie. *An Introduction to Modern Astrophyics*. Addison-Wesley Publishing Company, Inc., 1996.

[CT65]   J.W. Cooley and J.W. Tukey. An algorithm for the machine calculation of a complex fourier series. *Mathematics of Computation*, 19(90):297–301, April 1965.

[DD90]   Russell L. DeValois and Karen K. DeValois. *Spatial Vision*. Oxford University Press, August 1990.

[DNFP$^+$05]   S. De Nicola, A. Finizio, G. Pierattini, P. Ferraro, and D. Alfieri. Angular spectrum method with correction of anamorphism for numerical reconstruction of digital holograms on tilted planes. *Optics Express*, 13(24):9935–9940, November 2005.

[Ehr93]   James C. Ehrhardt. Hexagonal fast fourier transform with rectangular output. *IEEE Transactions on Signal Processing*, 41(3):1469–1472, March 1993.

[EO04]   G. B. Esmer and L. Onural. Simulation of scalar optical diffraction between arbitrarily oriented planes. In *First International Symposium on Control, Communications and Signal Processing*, pages 225–228, 2004.

[Fow89]   Grant R. Fowles. *Introduction to Modern Optics*. Dover Publications, Inc., 1989.

[FWS$^+$92]   K. Frohlich, U. Wagemann, J. Schulat, H. Schutte, and C.G. Stojanoff. Fabrication and test of a holographic concentrator for two color pv-operation. In V. Wittwer, C.G. Grangvist, and C.M. Lampert, editors, *Proc. SPIE Vol. 2255, Optical Materials Technology for Energy Efficiency and Solar Energy Conversion XIII*, pages 812–189, 1992.

[Gab48]   D. Gabor. A new microscopic principle. *Nature*, 161:777–778, May 1948.

[GAD⁺02]   Y. Gu, Z. Ansari, C. Dunsby, D. Parsons-Karavassilis, J. Siegel, M. Itoh, and P.M.W. French. High-speed 3d imaging using photorefractive holography with novel low-coherence interferometers. *Journal of Modern Optics*, 49(5):877–887, 2002.

[GFDN⁺01] S. Grilli, P. Ferraro, S. De Nicola, A. Finizio, G. Pierattini, and R. Meucci. Whole optical wavefields reconstruction by digital holography. *Optics Express*, 9(6):294–302, September 2001.

[GHTI98]   B. Grant, A. Helser, and R.M. Taylor II. Adding force display to a stereoscopic head-tracked projection display. In *1998 IEEE Virtual Reality Annual International Symposium*, pages 81–88, March 1998.

[GMD96]   Javier Garcia, David Mas, and Rainer G. Dorsch. Fractional-fourier-transform calculation through the fast-fourier-transform algorithm. *Applied Optics*, 35(35):7013–7018, December 1996.

[GR94]   I.S. Gradshteyn and I.M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, 1994.

[Gri89]   David J. Griffiths. *Introduction to Electrodynamics*. Prentice-Hall, Inc., 1989.

[Gri02]   Artyom M. Grigoryan. Efficient algorithms for computing the 2-d hexagonal fourier transforms. *IEEE Transactions on Signal Processing*, 50(6):1438–1448, June 2002.

[Har96]   P. Hariharan. *Optical Holography*. Cambridge University Press, 1996.

[HJB85]   M. T. Heideman, D. H. Johnson, and C. S. Burrus. Gauss and the history of the fast fourier transform. *Archive for History of Exact Sciences*, 34(3):265–277, September 1985.

[HMG03]   Michael Huebschman, Bala Munjuluri, and Harold R. Garner. Dynamic holographic 3-d image projection. *Optics Express*, 11(5):437–445, March 2003.

[HMHG05]   Michael L. Huebschman, Bala Munjuluri, Jeremy Hunt, and Harold R. Garner. Holographic video display using digital micromirrors. In Liang-Chy Chien, editor, *Proc. SPIE Vol. 5742, Emerging Liquid Crystal Technologies*, pages 1–14, April 2005.

[How95]   Ian P. Howard. *Binocular Vision and Stereopsis*. Oxford University Press, November 1995.

[HS03]   B. Hennelly and J.T. Sheridan. Optical image encryption by random shifting in fractional fourier domains. *Optics Letters*, 28(4):269–271, February 2003.

[HvSS⁺05] M. Hain, W. von Spiegel, M. Schmiedchen, T. Tschudi, and B. Javidi. 3d integral imaging using diffractive fresnel lens arrays. *Optics Express*, 13:315–326, 2005.

[IIS71]   Y. Ichioka, M. Izumi, and T. Suzuki. Scanning halftone plotter and computer-generated continuous-tone hologram. *Applied Optics*, 10(2):403–411, February 1971.

[IS04]   Tomoyoshi Ito and Tomoyoshi Shimobaba. One-unit system for electroholography by use of a special-purpose computational chip with a high-resolution liquid-crystal display toward a three-dimensional television. *Optics Express*, 12(9):1788–1793, May 2004.

[JFTC01]   Bahram Javidi, Yann Frauel, Enrique Tajahuerce, and Maria-Albertina Castro. Distortion invariant 3d object recognition using digital holography. In P. Ambs and F.R. Beyette, editors, *Proc. SPIE Vol. 4435, Wave Optics and VLSI Photonic Devices for Information Processing*, pages 23–30, December 2001.

[JNK+95]   M.W. Jones, G.P. Nordin, J.H. Kulick, R.G. Lindquist, and S.T. Kowel. Real-time three-dimensional display based on the partial pixel architecture. *Optics Letters*, 20:1418–1420, 1995.

[Joo86]   Georg Joos. *Theoretical Physics*. Dover Publications, Inc., 1986.

[Lap78]   Daniel Lapedes. *Dictionary of Scientific and Technical Terms: Second Edition*. McGraw Hill, 1978.

[LBB+05]   Michael Liebling, Thierry-Francois Bernhard, Adrian H. Bachmann, Luc Froehly, Theo Lasser, and Michael Unser. Continuous wavelet transform ridge extraction for spectral interferometry imaging. In Valery V. Tuchin, Joseph A. Izatt, and James G. Fujimoto, editors, *Proc. SPIE Vol. 5690, Coherence domain optical methods and optical coherence tomography in biomedicine IX*, pages 397–402, April 2005.

[LCR02]   Jacques Ludman, H. John Caulfield, and Juanita Riccobono. *Holography for the New Millennium*. Springer-Verlag, 2002.

[Les89]   Detlef Leseberg. Sizable fresnel-type hologram generated by computer. *Journal of the Optical Society of America A*, 6(2):229–233, February 1989.

[LF88]   Detlef Leseberg and Christian Frere. Computer-generated holograms of 3-d objects composed of tilted planar segments. *Applied Optics*, 27(14):3020–3024, July 1988.

[LHCW00]   S.H. Lin, K.Y. Hsu, W.Z. Chen, and W.T. Whang. Phenanthrenequinone-doped poly(methyl methacrylate) photopolymer bulk for volume holographic data storage. *Optics Letters*, 25:451, 2000.

[LIHD04]   Hongen Liao, Makoto Iwahara, Nobuhiko Hata, and Takeyoshi Dohi. High-quality integral videography using a multiprojector. *Optics Express*, 12(6):1067–1076, March 2004.

[Loh93]   Adolf W. Lohmann. Image rotation, wigner rotation, and the fractional fourier transform. *Journal of the Optical Society of America A*, 10(10):2181–2186, October 1993.

[LP67]   A.W. Lohmann and D.P. Paris. Binary fraunhofer holograms generated by computer. *Applied Optics*, 6:1739–1748, 1967.

[LU62]   E. N. Leith and J. Upatnieks. Reconstructed wavefronts and communication theory. *Journal of the Optical Society of America*, 52:1123–1130, 1962.

[LU64]   E. N. Leith and J. Upatnieks. Wavefront reconstruction with diffused illumination and three-dimensional object. *Journal of the Optical Society of America*, 54:1295–1301, 1964.

[Luc94]   Mark Lucente. *Diffraction-Specific Fringe Computation for Electro-Holography*. PhD thesis, Massachusetts Institute of Technology, September 1994.

51

[Luc96]   Mark Lucente. Holographic bandwidth compression using spatial subsampling. *Optical Engineering*, 35(6):1529–1537, June 1996.

[Mac92]   A.E. Macgregor. Computer generated holograms from dot matrix and laser printers. *American Journal of Physics*, 60(9):839–846, September 1992.

[MCJMCS05] M. Martinez-Corral, B. Javidi, R. Martnez-Cuenca, and G. Saavedra. Formation of real, orthoscopic integral images by smart pixel mapping. *Optics Express*, 13:9175–9180, 2005.

[MDM+05]   R. McLeod, A. Daiber, M. McDonald, T. Robertson, T. Slagle, S. Sochava, and L. Hesselink. Microholographic multilayer optical disk data storage. *Applied Optics*, 44:3197–3207, 2005.

[MGF+99]   David Mas, Javier Garcia, Carlos Ferreira, Luis M. Bernardo, and Francisco Marinho. Fast algorithms for free-space diffraction patterns calculation. *Optics Communications*, 164:233–245, June 1999.

[MJvR03]   Jurriaan D. Mulder, Jack Jansen, and Arjen van Rhijn. An affordable optical head tracking system for desktop vr/ar systems. In *2003 EGVE Proceedings of the workshop on Virtual environments*, pages 215–223. ACM Press, 2003.

[MK03]   K. Matsushima and A. Kondoh. Wave optical algorithm for creating digitally synthetic holograms of three-dimensional surface objects. In Tung H. Jeong and Sylvia H. Stevenson, editors, *Proc. SPIE Vol. 5005, Practical Holography XVII and Holographic Materials IX*, pages 190–197, May 2003.

[MO93]   David Mendlovic and Haldun M. Ozaktas. Fractional fourier transforms and their optical implementation: I. *Journal of the Optical Society of America A*, 10(9):1875–1881, September 1993.

[MRCD02]   Louis-Philippe Morency, Ali Rahimi, Neal Checka, and Trevor Darrell. Fast stereo-based head tracking for interactive environments. In *Proceedings of Conference on Automatic Face and Gesture Recognition*, 2002.

[MRM+05]   P. Marquet, B. Rappaz, P. Magistretti, E. Cuche, Y. Emery, T. Colomb, and C. Depeursinge. Digital holographic microscopy: a noninvasive contrast imaging technique allowing quantitative visualization of living cells with subwavelength axial accuracy. *Optics Letters*, 30:468–470, 2005.

[MSLC05]   M.F. Marmor, S. Shaikh, M.S. Livingstone, and B.R. Conway. Was rembrandt stereoblind? *New England Journal of Medicine*, 352:631–632, February 2005.

[MSW02]   Kyoji Matsushima, Hagen Schimmel, and Frank Wyrowski. New creation algorithm for digitally synthesized holograms in surface model by diffraction from tilted planes. In Stephen A. Benton, Sylvia H. Stevenson, and T. John Trout, editors, *Proc. SPIE Vol. 4659, Practical Holography XVI and Holographic Materials VIII*, pages 53–60, June 2002.

[MT00]   Kyoji Matsushima and Masahiro Takai. Recurrence formulas for fast creation of synthetic three-dimensional holograms. *Applied Optics*, 39(35):6587–6594, December 2000.

[MW71]   A.J. MacGovern and J.C. Wyant. Computer generated holograms for testing optical elements. *Applied Optics*, 10(3):619–624, March 1971.

[MZD+95]   David Mendlovic, Zeev Zalevsky, Rainer G. Dorsch, Yigal Bitran, Adolf W. Lohmann, and Haldun Ozaktas. New signal representation based on the fractional fourier transform: definitions. *Journal of the Optical Society of America A*, 12(11):2424–2431, November 1995.

[NOY+97]   Osamu Nishikawa, Takatsune Okada, Hiroshi Yoshikawa, Koki Sato, and Toshio Honda. High-speed holographic-stereogram calculation method using 2d fft. In Ivan Cindrich and Sing H. Lee, editors, *Proc. SPIE Vol. 3010, Diffractive and Holographic Device Technologies and Applications IV*, pages 49–57, May 1997.

[NYH01]   T. Naemura, T. Yoshida, and H. Harashima. 3-d computer graphics based on integral photography. *Optics Express*, 8:255–262, 2001.

[OBMO94]   Haldun M. Ozaktas, Billur Barshan, David Mendlovic, and Levent Onural. Convolution, filtering, and multiplexing in fractional fourier domains and their relation to chirp and wavelet transforms. *Journal of the Optical Society of America A*, 11(2):547–559, February 1994.

[OM93]   Haldun M. Ozaktas and David Mendlovic. Fractional fourier transforms and their optical implementation: Ii. *Journal of the Optical Society of America A*, 10(12):2522–2531, December 1993.

[OO05]   L. Onural and H.M. Ozaktas. Signal processing issues in diffraction and holographic 3dtv. In *Proc. EURASIP 13th European Signal Processing Conference*, 2005.

[OZK01]   Haldun M. Ozaktas, Zeev Zalevsky, and M. Alper Kutay. *The Fractional Fourier Transform with Applications in Optics and Signal Processing*. John Wiley and Sons, Inc., 2001.

[PA99]   Richard S. Pfisterer and Farzin Aghdasi. Hexagonal wavelets for the detection of masses in digitized mammograms. In Michael A. Unser, Akram Aldroubi, and Andrew F. Laine, editors, *Proc. SPIE Vol. 3813, Wavelet Applications in Signal and Image Processing VII*, pages 966–977, October 1999.

[PAIK05]   T.C. Poon, T. Akin, G. Indebetouw, and T. Kim. Horizontal-parallax-only electronic holography. *Optics Express*, 13(7):2427–2432, April 2005.

[Per96]   Senthil Periaswamy. Detection of microcalcifications in mammograms using hexagonal wavelets. Master's thesis, Government College of Technology, Coimbatore, India, 1996.

[PF94]   Pierre Pellat-Finet. Fresnel diffraction and the fractional-order fourier transform. *Optics Letters*, 19(18):1388–1390, September 1994.

[Pir02]   Luciano Pirodda. Time-integrated holography and optical tomography. *Optics Express*, 10(15):663–669, July 2002.

[PKK+04]   J. H. Park, Y. Kim, J. Kim, S. W. Min, and B. Lee. Three-dimensional display scheme based on integral imaging with three-dimensional information processing. *Optics Express*, 12:6020–6032, 2004.

[PTVF92]   William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992.

[Que04]   Tyeler Quentmeyer. Delivering real-time holographic video content with off-the-shelf pc hardware. Master's thesis, Massachusetts Institute of Technology, May 2004.

[RBD$^+$97] Alf Ritter, Thomas Benziger, Oliver Deussen, Thomas Strothotte, and Hubert Wagener. Synthetic holograms of splines. In H.P. Seidel, B. Girod, and H. Niemann, editors, *3D Image Analysis and Synthesis 1997*, pages 11–18, November 1997.

[Ric70]   W. Richards. Stereopsis and stereoblindness. *Experimental Brain Research*, 10(4):380–388, 1970.

[Roh94]   James William Rohlf. *Modern Physics from α to Z^0*. John Wiley and Sons, Inc., 1994.

[SHBL92]   Pierre St. Hilaire, Stephen A. Benton, and Mark Lucente. Synthetic aperture holography: a novel approach to three-dimensional displays. *Journal of the Optical Society of America A*, 9(11):1969–1977, November 1992.

[SIY04]   Yusuke Sando, Masahide Itoh, and Toyohiko Yatagai. Color computer-generated holograms from projection images. *Optics Express*, 12(11):2487–2493, May 2004.

[SJ94]   U. Schnars and W. Juptner. Direct recording of holograms by a ccd target and numerical reconstruction. *Applied Optics*, 33(2):179–181, January 1994.

[SR00]   Mark R.A. Shegelski and Timothy A. Rothwell. Rotational tomographic reconstruction of multiple in-line holograms. *Canadian Journal of Physics*, 78:1005–1016, 2000.

[SSMI05]   Tomoyoshi Shimobaba, Atsushi Shiraki, Nobuyuki Masuda, and Tomoyoshi Ito. Electroholographic display unit for three-dimensional display by use of special-purpose computational chip for holography and reflective lcd panel. *Optics Express*, 13(11):4196–4201, May 2005.

[Sto93]   J.L. Stone. Photovoltaics: Unlimited electric energy from the sun. *Physics Today*, 22, September 1993.

[TJ00]   Enrique Tajahuerce and Bahram Javidi. Encrypting three-dimensional information with digital holography. *Applied Optics*, 39(35):6595–6601, December 2000.

[Tre95]   Seymour Trester. Computer simulated holography and computer generated holograms. *American Journal of Physics*, 64(4):472–478, April 1995.

[VDVBU$^+$04] Dimitri Van De Ville, Thierry Blu, Michael Unser, Wilfried Philips, Ignace Lemahieu, and Rik Van de Walle. Hex-splines: A novel spline family for hexagonal lattices. *IEEE Transactions on Image Processing*, 13(6):758–772, June 2004.

[WkFB99]   Bernhard Weß kamp, Frank Fetthauer, and Olof Bryngdahl. Synthetic display holography: line segments through the hologram. *Journal of Modern Optics*, 46(10):1503–1512, 1999.

[WP84]   J.C. Weed and R.J. Polge. An efficient implementation of a hexagonal fft. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84*, volume 9, pages 488–491, March 1984.

[WZJW06]   Xiaogang Wang, Daomu Zhao, Feng Jing, and Xiaofeng Wei. Information synthesis (complex amplitude addition and subtraction) and encryption with digital holography and virtual optics. *Optics Express*, 14(4):1476–1486, February 2006.

[YKC⁺04]   Jun Yan, Stephen T. Kowel, Hyoung Jin Cho, Chong H. Ahn, Gregory P. Nordin, and Jeffrey H. Kulick. Autostereoscopic three-dimensional display based on a micromirror array. *Applied Optics*, 43(18):3686–3696, June 2004.

[YKCA01]   Jun Yan, Stephen T. Kowel, Hyoung Jin Cho, and Chong H. Ahn. Real-time full-color three-dimensional display with a micromirror array. *Optics Letters*, 26(14):1075–1077, July 2001.

[YKOM01]   Ichirou Yamaguchi, Jun-ichi Kato, Sohgo Ohta, and Jun Mizuno. Image formation in phase-shifting digital holography and applications to microscopy. *Applied Optics*, 40(34):6177–6186, December 2001.

[ZR00]   Jaime L. Zapata and Gerhard X. Ritter. Fast fourier transform for hexagonal aggregates. *Journal of Mathematical Imaging and Vision*, 12:183–197, 2000.

Vita

Richard Muffoletto was born in Baton Rouge, Louisiana, on 26 July, 1978, to Joe and Linda Muffoletto. He attended Baton Rouge Magnet High School and graduated in 1996. He graduated from Louisiana State University with bachelor degrees in physics and computer science in May 2001, and began pursuing a doctoral degree in the Department of Computer Science in August 2002 as a Board of Regents Fellow. In June 2002, Richard married Michele Robichaux, born in Raceland, Louisiana, to Clyde and Teresa Robichaux.