Senior Projects Holography Report

Ben Sterling, Dongkyu Kim, Junbum Kim, David Kim

2018-12-02

1 Introduction

Fast prototyping of real-world object has been addressed by many engineers and companies over the past few decades. For instance, 3D printing has especially gained much attention for its practicality. It has improved the time required to produce a visible and tangible product in a matter of few hours. On the other end of the spectrum, holography has been attracting attention due to its potentials. The technology, compared to 3D printing, can produce less tangible but faster results. Moreover, holography-based 3D printing is much faster, i.e. matter of seconds instead of hours, than the traditional 3D printer that builds one microscopic layer at a time.[1] These characteristics make holography technology a crucial tool in 3D modelling on its own and also in context of 3D printing.

However, the application still remains a mystery because of its barriers to entry - not only the parts required for the technology are too expensive for easy access, but the theory behind it requires sophisticated understanding of physics. To make things worse, unlike 3D printing, the theory is essential in crafting a reliable holography projection. We believe these traits make holography an adequate topic for our senior project; it is very interesting, and successful investigation of the technology will be a great influence on the scientific community. In this paper, we will be exploring 3D modelling and holographic projection to demystify and familiarize the audience with the concept of holography so that more people could investigate and invest in the technology.

The following sections of the paper are organized by discussing the experimental procedures to produce a relatively affordable holographic image of a real-life object. The main sections of the paper include the sampling stage, digital conversion stage, transmission stage, and optical holography projection stage. The sampling stage scans a real-world object and stores it digitally on a computer. The digital conversion stage transforms the digitally stored data into a holographic compatible format. The transformed image is then transmitted into the holographic machine called Source Light Modula-

tor (SLM). Finally, the optical setup projects the holographic image created by the SLM onto a flat white surface at the projection stage. Each of the stages is explained in more detail in the following sections of the paper.

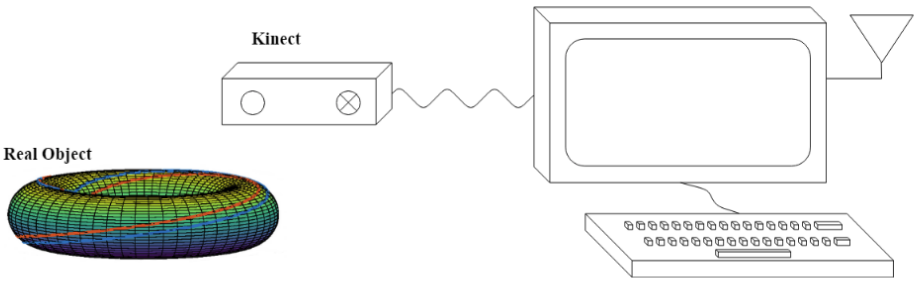


Figure 1: Sampling object using Kinect

2 Sampling Stage

The first stage of the project is sampling a real-world object and storing it into a computer digitally. Since the perception of depth is a crucial part of the holographic projection, it is imperative to sample both the shape and depth of a real-time object. In this experiment, Xbox Kinect for Windows Development was used to sample objects. Kinect is originally developed as a module for Xbox to track human motions for entertainment purposes. It can capture an object from a single viewpoint and reconstruct 3D models based on it because it has an ordinary camera, a depth sensor, and an infrared sensor.

Windows Presentation Foundation (WPF) application was built to communicate the hardware with software. The program was developed using C# which is the natural platform for the machine. Kinect Studio Version 2.0 and Kinect Software Development Kits were installed to develop the program. Specifically, the KinectFusionExplorer module proved practical for the purpose. The module allows 3D modelling from a single photo shoot from one angle by reconstructing picture from the ordinary camera, infrared sensors, and depth sensors. Although the technology cannot provide a 360-degree capture of the sample, it does reconstruct the sides of the image with

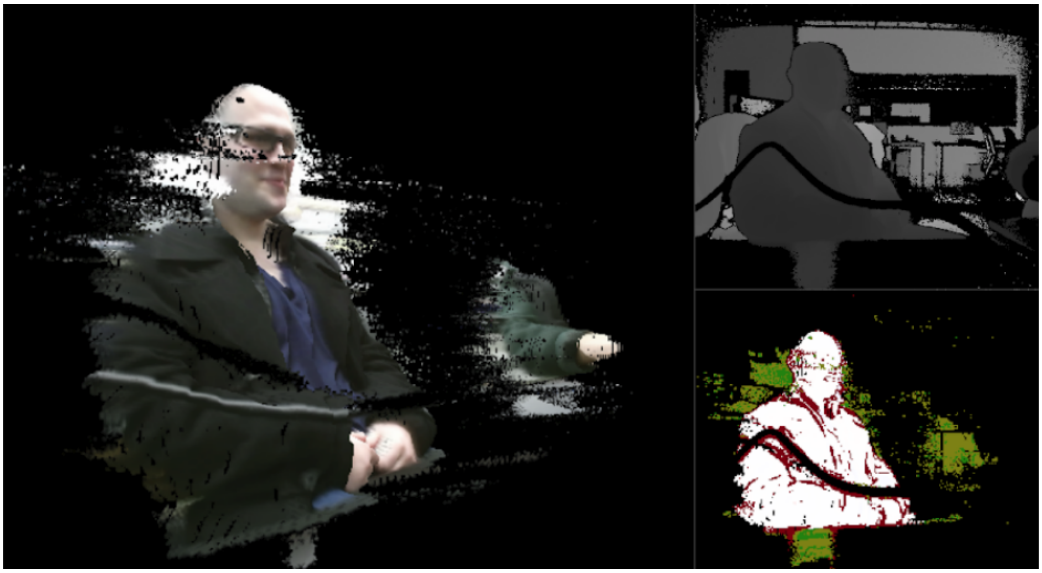


Figure 2: Sample data obtained from Kinect

only small depth errors. The rotation of the model allows users to see the sides of the real object within the computer. Note that USB 3.0 is required to reproduce the program, as the Kinect hardware requires the extra bandwidth for communication of data. Features such as shape and depth collected from the Kinect is stored in the computer to later be transformed. More details in what information needs to be encrypted into holographic image remains a subject of investigation at this point. Refer to Figure 2 for a 3D sample constructed using a depth sensor (top right) and an infrared sensor (bottom right).

The WPF application serves two important purposes. Firstly, it provides a user-friendly interface so that the users can use the application without much difficulty. The users can select the thresholds at which they want to set their foreground and background of the sampled data. These would allow users to produce a holographic image without having to know the details of the functionality. Easily speaking, it serves as a black box for the end users. Secondly, it allows the integration of different programming modules that we

need for the program. Specifically, the data extracted from the Kinect needs to be converted into a proper format, and be sent to the SLM. There are existing libraries to transmit data to SLM in python. C# WPF applications allow users to execute python script on top of the application. To summarize, the application serves as an easy way to integrate software parts while hiding unnecessary details from the end users.

3 Digital Conversion Stage

The first program that has to run on top of the WPF application transforms the image obtained from the previous sample stage to an SLM compatible format. The system that we have is JD9554 spatial light modulator, which is part of Jasper Display's EDKv2 education kit. This modulator is a phase-only modulator, hence our program must convert the image into a phase mask that contains enough information to reconstruct an image. There are multiple ways of achieving this result, but one of the most efficient and popular approaches for our application is the kinoform method. The kinoform method, which is based on the observation that when the actual projection occurs, there are random-phase masks applied to the object pattern, making the amplitude of the actual object pattern unimportant. This observation leads to the kinoform method, which becomes the basis of our algorithm. According to Poon, the kinoform method without any additional adjustment is very noisy due to its dimension reducing nature [2]. In order to optimize the phase-only holograms, iterative Fourier transform algorithm (IFTA) is used. This algorithm iteratively reduces the error of the reconstructed image by smoothing out the complex amplitude.

Fig. 3 shows changes in the root mean square error (RMSE) as more iterations are done using the IFTA algorithm. There is a big drop in the error from iteration 1 to 4, and a plateau from 4 to 10, and then the error starts to decay again. We have noticed that each iteration takes about 0.1 seconds, and for our real-time application, doing only 4 loops of IFTA algorithm would be the most beneficial case, as it delivers a reasonable quality in the shortest time. Fig. 4 contains images regarding our experiment with the algorithm. We have used the "peppers.png" picture for our original image and obtained the phase mask for the SLM. The example that is given in the figure was obtained after 20th iteration. It is important to note that this will be the image that will be transferred to the SLM and used to steer the incoming lights to create the hologram. As we do not have a physical system yet, the images were reconstructed mathematically. Reconstructed images after 2nd

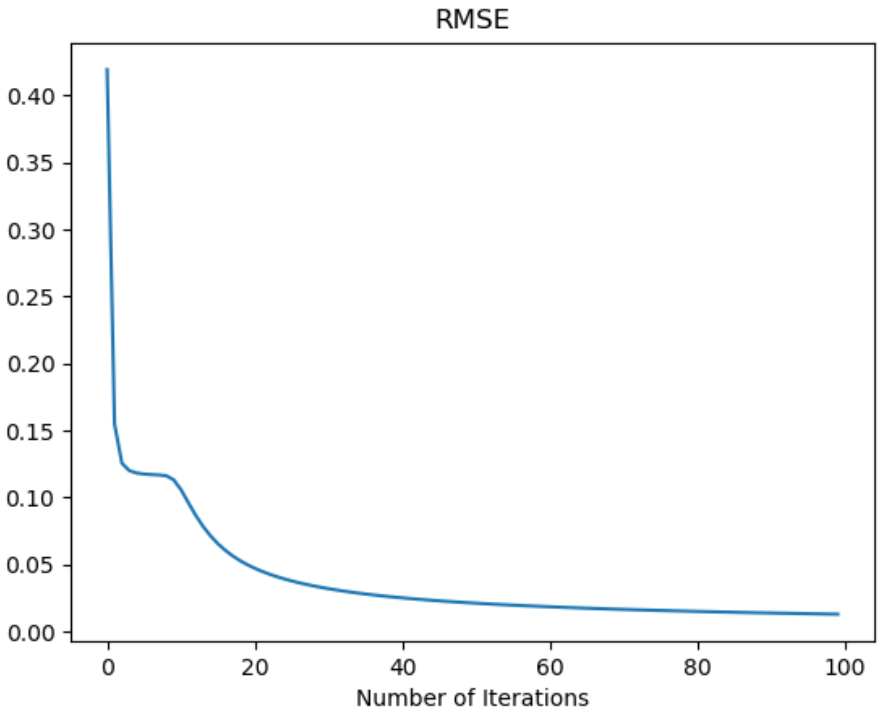


Figure 3: Root Mean Square Error (RMSE) of the Reconstructed Image to the Original Image

iteration and 20th iteration were compared, and there is a visible quality improvement for the latter.

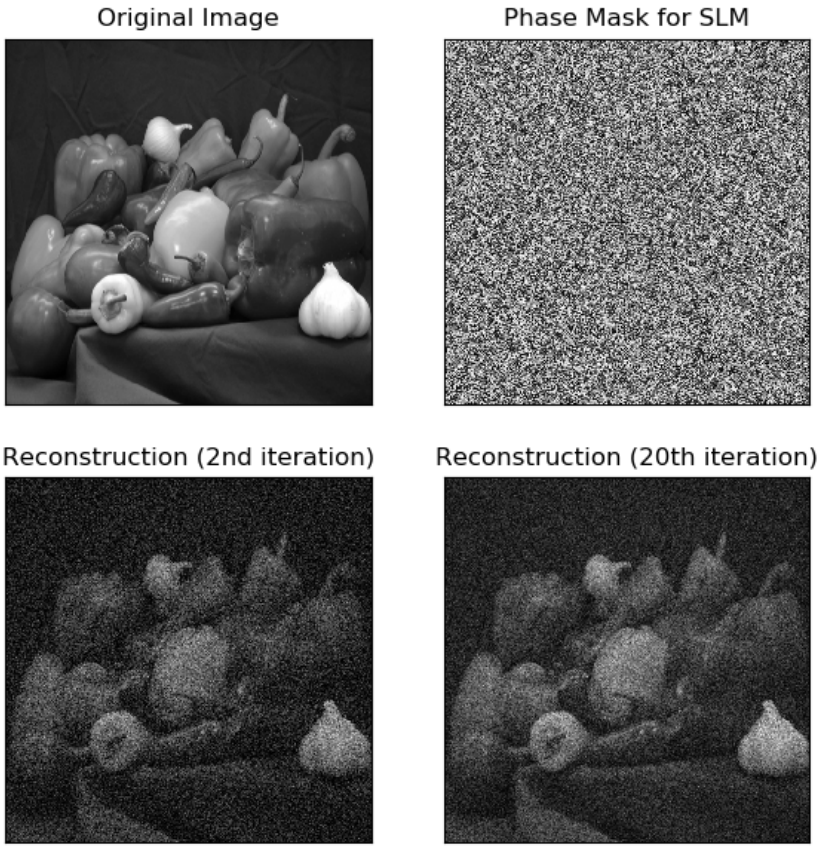


Figure 4: Collection of Images for the Performance Measurement

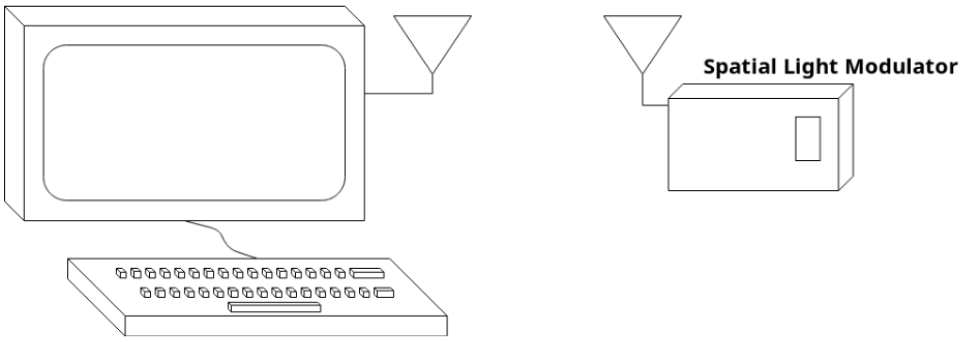


Figure 5: Transmitting data from Computer to SLM

4 Digital Projection Stage

In the most basic form, SLM can be considered as a monitor that displays image obtained from a computer. The SLM model that we use connects to the computer via an HDMI port. For the manipulation of the image shown on the SLM screen, we use slmPy, a python module created to dynamically control the SLM by Sebastien Popoff [3]. The depth and phase information of the image produced from the digital conversion stage can be sent to the SLM as numpy arrays using the slmPy module. The module has a function that retrieves the resolution of the screen, which may be critical for successful projection of the holography. The depth and phase information is then displayed on the reflective surface of the SLM, which simultaneously is shined by a laser that goes through a beam splitter. The laser that reflects off of the surface will contain the depth and phase information of the image, which subsequently will be projected on a 2D screen as holography.

5 Holography Background

5.1 Wave Fundamentals

The fundamentals of Holography start from Maxwell's Equations. In the most general case, they are formulated as follows:

$$\nabla \cdot \vec{D} = \rho \tag{1}$$

$$\nabla \cdot \vec{B} = 0 \tag{2}$$

$$\nabla \times \vec{H} = \vec{J} + \frac{\partial \vec{D}}{\partial t} \tag{3}$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{4}$$

However, we are interested in utilizing these equations to study optics. Since light waves travel in the air, $\rho = 0$ and $\vec{J} = \vec{0}$ because there is no source charge or current in the air. Here are the equations we obtain setting sources to zero:

$$\nabla \cdot \vec{D} = 0 \tag{5}$$

$$\nabla \cdot \vec{B} = 0 \tag{6}$$

$$\nabla \times \vec{H} = \frac{\partial \vec{D}}{\partial t} \tag{7}$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{8}$$

Lastly, we want to relate D to E, and B to H. The most general relationships that describes these quantities are:

$$\vec{D} = \epsilon_0 \vec{E} + \vec{P} \tag{9}$$

$$\vec{B} = \mu_0(\vec{H} + \vec{M}) \tag{10}$$

We assume that all the materials we are working in are Linear, Homogeneous, and Isotropic. The last two equations can be simplified to:

$$\vec{D} = \epsilon \vec{E} \tag{11}$$

$$\vec{B} = \mu \vec{H} \tag{12}$$

where $\epsilon = \epsilon_r \epsilon_0$ and $\mu = \mu_r \mu_0$ are constants. If we combine the last two equations along with Maxwell's Equations we get:

$$\nabla \cdot \vec{E} = 0 \tag{13}$$

$$\nabla \cdot \vec{H} = 0 \tag{14}$$

$$\nabla \times \vec{H} = \epsilon \frac{\partial E}{\partial t} \tag{15}$$

$$\nabla \times \vec{E} = -\mu \frac{\partial H}{\partial t} \tag{16}$$

then we utilize the following identity from vector calculus:

$$\nabla^2 F = \nabla(\nabla \cdot F) - \nabla \times \nabla \times F \tag{17}$$

and get:

$$\nabla^2 E - \mu\epsilon \frac{\partial^2 E}{\partial t^2} = 0 \tag{18}$$

$$\nabla^2 H - \mu\epsilon \frac{\partial^2 H}{\partial t^2} = 0 \tag{19}$$

The Multi-Dimensional Fourier Transform is defined as follows:

$$\hat{f}(\vec{k}) = \iint_{R^n} f(\vec{x}) e^{-j\vec{k}\cdot\vec{x}}\, d\vec{x} \tag{20}$$

and a plane wave is defined as follows:

$$\vec{E} = \vec{E_0} e^{j\omega t - j\vec{k}\cdot\vec{x}} \tag{21}$$

Because E and H field are functions of x, y, z, and time, it is possible to span all E and H fields with a superposition of plane waves. This motivates the notation:

$$\vec{E} = \vec{E_0}\psi \tag{22}$$

where

$$\psi = e^{j\omega t - j\vec{k}\cdot\vec{x}} \tag{23}$$

Thus, in ψ notation, the wave equation turns into:

$$\nabla^2\psi - \mu\epsilon\frac{\partial^2\psi}{\partial t^2} = 0 \tag{24}$$

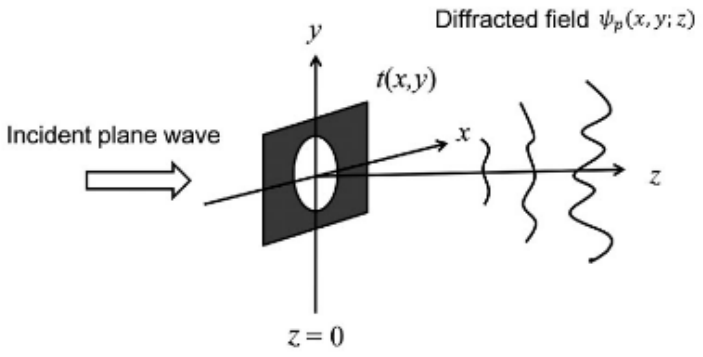
5.2 Plane and Spherical Wave Applications

For a single frequency, we are interested in the complex amplitude, ψ_p we define as follows:

$$\psi = \psi_p e^{j\omega t} \tag{25}$$

We can think of this quantity like the phasor representation of ψ.

The central application of the classical theory is to develop the math behind Fresnel Diffraction. An aperture can be visualized by this diagram:



We also define k_0 as the wave number in free space, such that

$$\nabla^2 \psi_p + k_0^2 \psi_p = 0 \tag{26}$$

since

$$\lambda f = c \rightarrow k_0 = \omega / c \tag{27}$$

By considering the incoming plane wave as a superposition of plane waves (like we did with the Multi-Dimensional Fourier Transform), we obtain the result:

$$\mathscr{F}\left\{ \frac{\partial^2 \psi_p}{\partial x^2} \right\} = (-jk_x)^2 \Psi_p = -k_x^2 \Psi_p \tag{28}$$

and

$$\mathscr{F}\left\{ \frac{\partial^2 \psi_p}{\partial y^2} \right\} = (-jk_y)^2 \Psi_p = -k_y^2 \Psi_p \tag{29}$$

where

$$\Psi_p = \mathscr{F}\{\psi_p\} \tag{30}$$

Finally, if we take the Fourier Transform of both sides of (26) we obtain:

$$\frac{\partial^2 \Psi_p}{\partial z^2} + k_0^2 \left(1 - \frac{k_x^2}{k_0^2} - \frac{k_y^2}{k_0^2} \right) \Psi_p = 0 \tag{31}$$

While this looks complicated, it is really only an Ordinary Differential Equation of Ψ_p with respect to z. It's solution is:

$$\Psi_p = \big(\Psi_p|_{z=0}\big) exp\bigg(- jk_0 z \sqrt{1 - \frac{k_x^2}{k_0^2} - \frac{k_y^2}{k_0^2}}\bigg) \tag{32}$$

If we let

$$\Psi_{p0} = \big(\Psi_p|_{z=0}\big) \tag{33}$$

Then we can recognize the solution in (32) as a transfer function. We define:

$$\mathscr{H} = \frac{\Psi_p}{\Psi_{p0}} = exp\bigg(- jk_0 z \sqrt{1 - \frac{k_x^2}{k_0^2} - \frac{k_y^2}{k_0^2}}\bigg) \tag{34}$$

$\mathscr{H}$ is called the Spatial Frequency Transfer Function of Propagation. It applies to our specific apparatus under consideration and describes Fresnel Diffraction of the output of a plane wave hitting an aperture. ψ_{p0} is the function controlled by the shape of the aperture; if this equals $\delta(x, y)$, for example, then the output of the aperture is purely the Spatial Frequency Transfer Function response, because this is equivalent to convolving with a delta in 2-Dimensions, returning $\mathscr{F}^{-1}\{\mathscr{H}\}$ in the time domain.

5.2.1 Fresnel Diffraction

We have $\mathscr{H}$, but more often than not, the traveling waves make very small angles from the normal direction of the aperture. This allows us to utilize the Paraxial Approximation:

$$\sqrt{1 - \frac{k_x^2}{k_0^2} - \frac{k_y^2}{k_0^2}} \approx 1 - \frac{k_x^2}{2k_0^2} - \frac{k_y^2}{2k_0^2} \tag{35}$$

Once we apply this approximation, the Spatial Frequency Transfer Function of Propagation simplifies:

$$\mathscr{H} = exp(-jk_0z)exp\left[\frac{j(k_x^2 + k_y^2)z}{2k_0}\right] \tag{36}$$

In this specific example, we are using the Two Dimensional Fourier Transform (z remains constant):

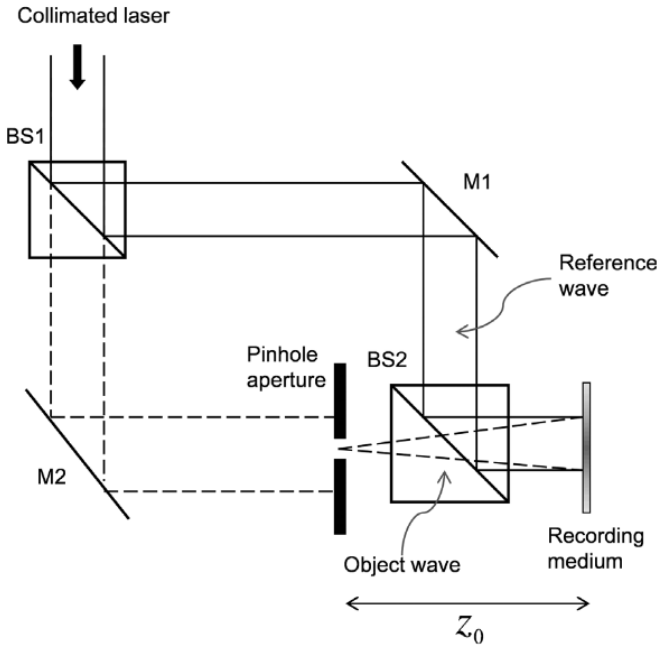
$$\Psi_p(k_x, k_y, z) = \iint_{R^2} \psi_p(x, y, z)e^{-jk_xx - jk_y}\,dx\,dy \tag{37}$$

Thus we get the Spatial Impulse Response, h:

$$h(x, y, z) = \mathscr{F}^{-1}\{\mathscr{H}\} = \frac{jk_0}{2\pi z}exp(-jk_0z)exp\left[\frac{-jk_0(x^2 + y^2)}{2z}\right] \tag{38}$$

5.3 Holography Fundamentals

The simplest holography setup, and a good starting point to understanding the subject, is this collection of Fresnel Plates:



First, we use a collimated laser so that the direction of propagation of each wave is parallel. The first beam splitter (BS1) divides the laser in two paths of equal power, one will be used as a reference and the other contains the object information. We recombine the waves at the second beam splitter (BS2). If the object wave is denoted as ψ_0 and the reference wave is ψ_r. Since our aperture is a pinhole, our input function (referencing the Spatial Frequency Transfer Function) is $\delta(x, y)$, thus:

$$\begin{aligned} \psi_0(x, y, z_0) &= \delta(x, y) * h(x, y, z_0) = h(x, y, z_0) \\ &= exp(-jk_0 z_0) \frac{jk_0}{2\pi z_0} exp\left[\frac{-jk_0(x^2 + y^2)^2}{2z_0}\right] \end{aligned} \tag{39}$$

and ψ_r is just a plane wave propagating in the z_0 direction:

$$\psi_r = a\, exp(-jk_0 z_0) \tag{40}$$

It is important to note that the light intensity is proportional to square of the complex amplitude, I(x,y) $\propto |\psi_p(x, y)|^2$. Thus we define transmittance, t(x,y) as:

$$t(x, y) = |\psi_p(x, y)|^2 \tag{41}$$

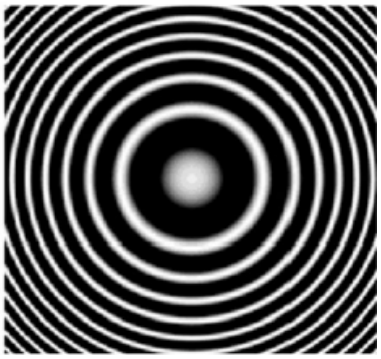
For our specific case:

$$\begin{aligned} t(x, y) &= |\psi_o(x, y) + \psi_r(x, y)|^2 = (\psi_o + \psi_r)(\psi_o + \psi_r)^* \\ &= a^2 + (\frac{k_0^2}{2\pi z_0})^2 - \frac{-jk_0 a}{2\pi z_0} exp\left[\frac{jk_0}{2z_0}(x^2 + y^2)\right] + \\ &\qquad \frac{-jk_0 a}{2\pi z_0} \exp\left[\frac{-jk_0}{2z_0}(x^2 + y^2)\right] \end{aligned} \tag{42}$$

We can recognize the sine function inside the last line:

$$t(x, y) = a^2 + \left(\frac{k_0}{2\pi z_0}\right)^2 + \frac{ak_0}{\pi z_0} sin\left[\frac{k_0}{2z_0}(x^2 + y^2)\right] \tag{43}$$

And thus, we can observe the diffraction pattern of this aperture as follows:



By inspection, the output of the machine obeys a sinusoidal pattern, with some DC value in towards the center (x = 0 and y = 0), and the frequency of the sinusoid increases parabolically moving radially outward. Also, for any tangent vector v_p, the spatial frequency is defined as the directional derivative of the angle:

$$v_p\left[\frac{k_0(x^2+y^2)}{2z_0}\right] = \frac{k_0}{z_0}(v_1p_1 + v_2p_2), \forall v_p \in T_p\mathbb{R}^2 \tag{44}$$

This result is more significant in differential form:

$$d\left(\frac{k_0(x^2+y^2)}{2z_0}\right) = \frac{k_0}{z_0}(xdx + ydy) \tag{45}$$

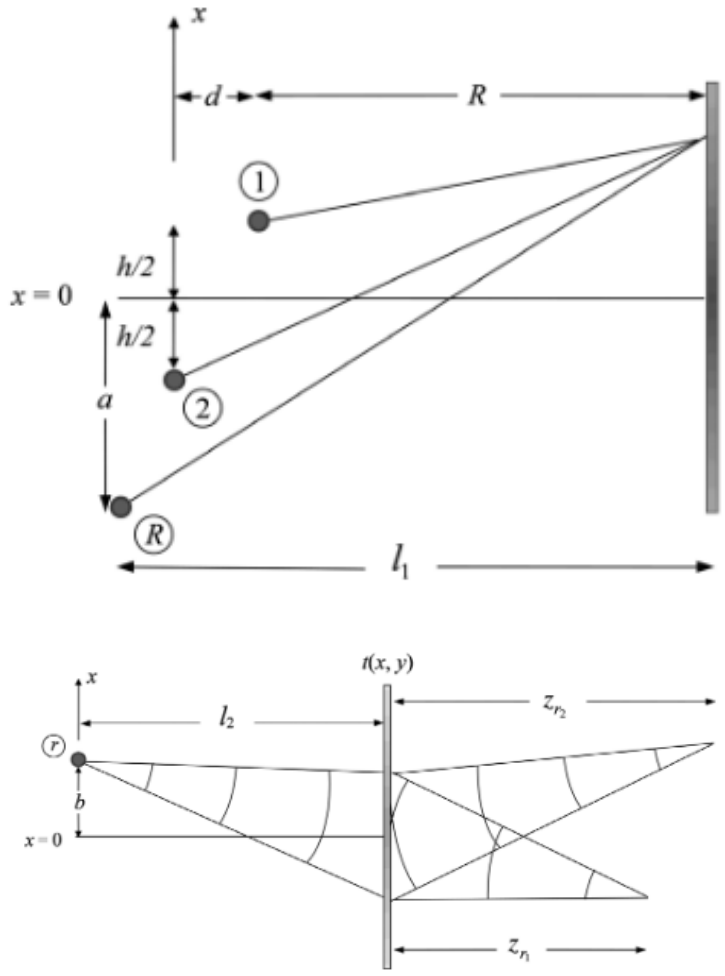
The frequency increases linearly in any direction of our choosing. It is important to note that an observer at a distance will see the following ψ_{rec} instead because they are at a distance, z, away from the recording medium.

$$\psi_p = \psi_{rec}t(x,y) * h(x,y,z) \tag{46}$$

Where ψ_{rec} is the field on the projecting surface.

5.3.1 3D Holographic Imaging

In this next section, we wish to identify the effects of different co-planar displacements (x and y) and different distances from the medium (z). We can visualize this problem with the following diagrams:



The first graph displays two source points with a reference wave R, and the second displays the reconstructed point r (the location a viewer SEE'S the object).

The following identity under convolution helps to simplify the upcoming calculations.

$$\delta(x-\alpha, y-\beta) * h(x, y, z) = \delta(x, y) * h(x-\alpha, y-\beta, z) \tag{47}$$

The physical significance of this identity is that being off-center from the pinhole aperture is equivalent to translating the center Spatial Impulse

Response: a very important property.

Thus, we can find the three relevant complex fields as follows:

$$\psi_{p1} = \delta(x - \frac{h}{2}, y) * h(x, y, R) = h(x - \frac{h}{2}, y, R) \tag{48}$$

$$\psi_{p2} = \delta(x + \frac{h}{2}, y) * h(x, y, R + d) = h(x - \frac{h}{2}, y, R + d) \tag{49}$$

$$\psi_{pR} = \delta(x + a, y) * h(x, y, l_1) = h(x + a, y, l_1) \tag{50}$$

Now, the intensity can be found as follows (by definition)

$$\begin{aligned} t(x, y) = |\psi_{p1} + \psi_{p2} + \psi_{pR}|^2 \\ (\psi_{p1} + \psi_{p2} + \psi_{pR})(\psi_{p1} + \psi_{p2} + \psi_{pR})^* \end{aligned} \tag{51}$$

Each $\psi_{pi}\psi_{pi}^*$ for $i \in \mathbb{Z}^+$ is not very interesting, because the wave parts cancel and they are just zeroth order waves. Also, note $\psi_{pi}\psi_{pj}^* = (\psi_{pj}\psi_{pi}^*)^*$, so we only need three transmittances to determine the intensity

If we take one transmittance:

$$t_{rel1}(x, y) = \psi_{p1}^*\psi_{pR}(x, y) = exp(\frac{jk_0}{2R}[(x - \frac{h}{2})^2 + y^2])exp(\frac{-jk_0}{2R}[(x + a)^2 + y^2]) \tag{52}$$

and then solve for the reflected ψ, we get:

$$\psi_{pr}(x, y)t_{rel1}(x, y) * h(x, y, z) \tag{53}$$

Where:

$$\psi_{pr} = \delta(x - b, y) * h(x, y, l_2) = h(x - b, y, l_2) \tag{54}$$

This must be true because INDEPENDENT of the points 1 and 2, the field is being affected by a plane wave coming from the reconstruction point. Thus, if we want to approximate the reflected wave, it is given by:

$$\psi_{ref} \propto exp\bigg(\frac{-jk_0}{2l_2}[(x-b)^2+y^2]\bigg)exp\bigg(\frac{-jk_0}{2R}[(x-\frac{h}{2})^2+y^2]\bigg) \\ exp\bigg(\frac{-jk_0}{2l_1}[(x+a)^2+y^2]\bigg) * \frac{jk_0}{2\pi z}exp\bigg(\frac{-jk_0}{2z}[x^2+y^2]\bigg) \tag{55}$$

At this point in the derivation, we must remember the definition of 2D convolution we are working with:

$$\big(f_1 * f_2\big)(x,y) = \iint\limits_{R^2} f_1(x',y') f_2(x-x', y-y')\, dx'\, dy' \tag{56}$$

By far the most confusing part of this derivation: the coefficients in front of x'^2 and y'^2 must equal zero because the convolution in (55) must evaluate to a δ function. Think of it this way: If it did not evaluate to a δ, this would imply the field has multiple non-zero space values for a single image point; we do not allow a single point to spread into a cloud, this would violate our empirical observations.

Thus by inspection, we derive the equation:

$$\frac{1}{R} - \frac{1}{l_1} - \frac{1}{l_2} - \frac{1}{z_{r1}} = 0 \tag{57}$$

Through very tedious math, we get that:

$$\psi_{ref} \propto \delta\bigg[x + z_{r1}\bigg(\frac{b}{l_2} - \frac{h}{2R} - \frac{a}{l_1}\bigg), y\bigg] \tag{58}$$

Similarly for the second point:

$$\frac{1}{R+d} - \frac{1}{l_1} - \frac{1}{l_2} - \frac{1}{z_{r2}} = 0 \tag{59}$$

$$\psi_{ref} \propto \delta\bigg[x + z_{r2}\bigg(\frac{b}{l_2} - \frac{h}{2(R+d)} - \frac{a}{l_1}\bigg), y\bigg] \tag{60}$$

These equations are the starting point to understanding magnification and translation distortion of images. For the most part, we are simplifying the problem of displaying an image to effects between two points.

5.3.2 Holographic Magnification

In this context, magnification is defined as:

$$M^r_{Long} = \frac{z_{r2} - z_{r1}}{d} \tag{61}$$

This is literally a ratio of the hologram depth over the real depth between these two points.

We substitute from equation (57) and (59) (Assuming: R $\gg$ d)

$$M^r_{Long} = \left(\frac{l_1 l_2}{l_1 l_2 - R l_1 - R l_2} \right)^2 \tag{62}$$

We define lateral magnification as the difference in x virtual positions over the real x distance (h). We use equations (58) and (60):

$$\begin{aligned} M^r_{Lat} &= \frac{z_{r2}\left(\frac{b}{l_2} + \frac{h}{2(R+d)} - \frac{a}{l_1}\right) - z_{r1}\left(\frac{b}{l_2} - \frac{h}{2R} - \frac{a}{l_1}\right)}{h} \\ &\approx \frac{(z_{r2} - z_{r1})\left(\frac{b}{l_2} - \frac{a}{l_1}\right) + (z_{r2} + z_{r1})\frac{h}{2R}}{h} \end{aligned} \tag{63}$$

assuming $R \gg d$

If we align the points in a special way such that:

$$\frac{b}{l_2} - \frac{a}{l_1} = 0 \tag{64}$$

We get the fascinating result that

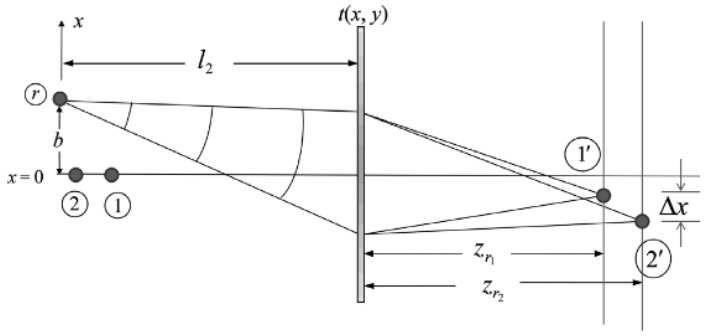
$$M^r_{Long} = (M^r_{Lat})^2 \tag{65}$$

5.3.3 Holographic Translation

The equation for Lateral Magnification gives us the distortion between two holographic points if (64) is not met:

$$\Delta x = x_1 - x_2 = (z_{r2} - z_{r1})\left(\frac{b}{l_2} - \frac{a}{l_1}\right) \tag{66}$$

This figure shows how an artificial Δx comes about from erroneous orientation

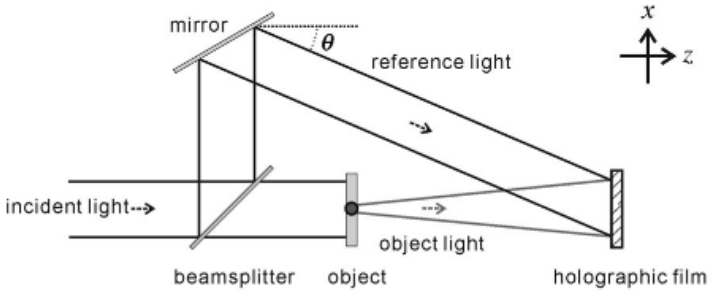


5.3.4 Off-Axis Holography

The standard holography setup we have studied is called on-axis holography. The only problem with this approach is that we get twin images when we expand the full equation for transmittance: (the conjugation terms make the twin images)

$$t(x, y) = |\psi_r + \psi_o|^2 = |\psi_r|^2 + |\psi_o|^2 + \psi_r^* \psi_o + \psi_r \psi_o^* \tag{67}$$

Off-Axis holography takes care of this problem by introducing the reference light at an angle θ



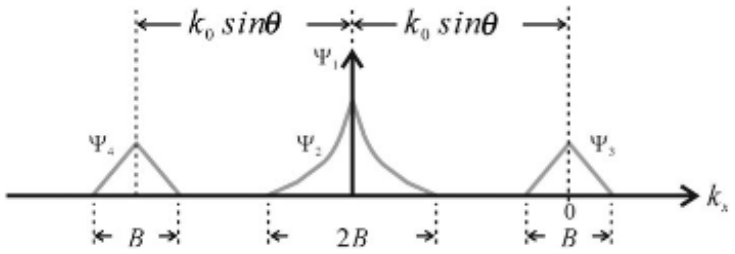
We can redo the logic of equation (67) as follows (the extra phase factor comes out from the extra distance the reference wave needs to travel)

$$t(x,y) = |\psi_r e^{jk_0 sin\theta x} + \psi_o|^2 = |\psi_r|^2 + |\psi_o|^2 + \psi_r^* \psi_o e^{-jk_0 sin(\theta)x} + \psi_r \psi_o^* e^{jk_0 sin(\theta)x}$$
$$= |\psi_r|^2 + |\psi_o|^2 + 2|\psi_0^* \psi_r| cos(2\pi f_x x + \phi(x,y)) \tag{68}$$

The phase getting scaled by $xsin(\theta)$ makes sense because at $\theta = \pi/2$, the extra distance turns into just x, and at $\theta = 0$ the reference wave is already in phase with the object wave, so no adjustment would be necessary. Just like the earlier section, we recognize the cosine identity in the top of the last equation (the ϕ is any extra phase that came from the ψ_o ψ_r pair not counting the angle θ). Specifically, $f_x = \frac{sin(\theta)}{\lambda}$.

Finally, we need to factor in the reconstruction light. If it has the same magnitude and direction as the reference light, we can represent it with the following wave function (which later gets convolved with h(x,y,z) to render the real image):

$$\begin{aligned} \psi &= \psi_r e^{jk_0 sin(\theta)x} t(x,y) \\ &= |\psi_r|^2 \psi_r e^{jk_0 sin(\theta)x} + |\psi_0|^2 \psi_r e^{jk_0 sin(\theta)x} \\ &\quad + \psi_o |\psi_r|^2 + \psi_o^* \psi_r^2 e^{2jk_0 sin(\theta)x} \end{aligned} \tag{69}$$



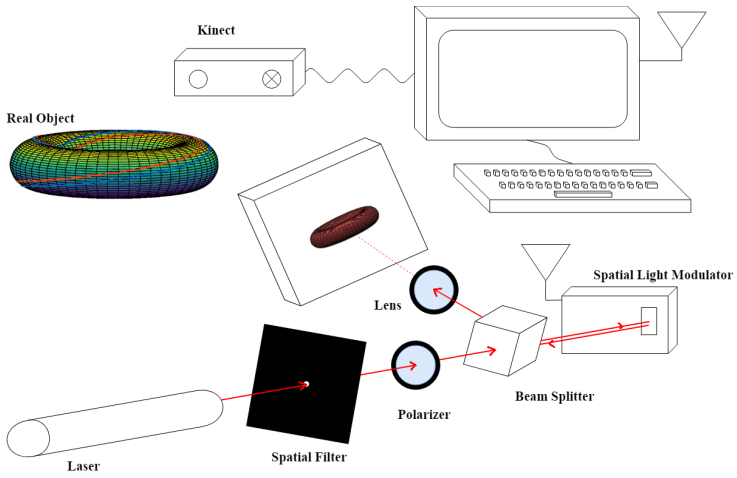
Upon closer examination, we can regard $k_0 sin(\theta)$ our carrier frequency (in radians). Since ψ_r is a reference light, it is entirely DC, so the first term in (69) is Ψ_1, the carrier. The second term in (69) is Ψ_2 because the

phasor convolves the magnitude-squared of the object wave out to the carrier frequency. The third (and most important) term, Ψ_3 is the only one that is not modulated by the carrier (it gets cancels out by the algebra). This is the only term that is visible to the human eye!

By inspection, the minimal angle θ we need to prevent aliasing can be given by:

$$k_0 sin(\theta) \geq \frac{3B}{2} \tag{70}$$

6 Project Setup



This is a picture of our Hollistic project. the software component is on top as described before. We start on the bottom left corner with a coherent laser source; it is important that the laser is coherent otherwise random phase artifacts would be introduced from the start further degrading the quality of the final hologram. We next use a pinhole apparatus to filter out the higher order noise artifacts that might have been introduced by the laser and surrounding distance; this also centers the laser by converting the incoming plane wave of the laser into a spherical wave (which turns into a plane wave again after applying the Paraxial Approximation). We also put a polarizer

in front of the pinhole so that the light incoming into the SLM has the anticipated linear polarization that it expects (the wrong polarization may not reflect off the SLM as expected).

We put a beam splitter in front of the polarizer to send half of the incoming beam onto the screen and half to the SLM. The SLM is electronically steered by the incoming video signal, and has 8 bit precision in the phase of the wave that it reflects. An important note is that the wave reflected off the SLM occupies the same physical space as the incoming wave: this works due to Transmission Line Theory (the two waves do not interfere with one another; they are governed only by Maxwell's Equations). Finally, the reflected wave goes into the beam splitter and sums with the incident light as both waves travel to the recording medium. This is how our hologram appears on the screen.

7 List of Materials

We are using a JD9554 Spatial Light Modulator, a P20S 20 micro-meter pinhole from Thorlabs, a BS013 Cube Beam Splitter, AL2520-A collimated lenses, C560TME-B Aspheric Lens, an LPVISE200-A Polarizer, and a high powered laser (we are using a low powered one at the moment, but may need to get the high powered one from Professor Yecko).

8 Status

Our progress of the project so far was greatly hindered by materials that we need not arriving due to the Canada Post strike. However, we have started setting up the system at the lab, and we feel confident that the hologram system would work with our knowledge. On the software side of the project, the program that converts images into inputs appropriate for the spatial light modulator (SLM) is implemented in two different algorithms. In addition, the Windows Presentation Foundation application that guides the transportation

of the Microsoft Kinect images or videos into the SLM is implemented and waiting for testing.

It has been tested that the digital projection stage of the SLM assisted by slmPy module works flawlessly on a computer monitor. The desired image in the form of numpy array is automatically resized to fit the entire screen, and both still image and animation are displayed without any issue. However, additional testing on the SLM is yet to be done as the setup was not completed for projection.

9 Future goals

For the upcoming semester, as all the materials have finally arrived, we will be building the holography system, and starting the necessary tests to pick the optimal algorithms for image conversion, resolutions, and sample methods. We hope to be done with the building of the system and getting a complete hologram image by late February so that we can concentrate on making the holography to look more 3D like, and presented in real-time. Currently, another missing piece of the system is the program that encodes the depth information appropriately into the 2D images. Our system inherently deals with 2D images, and the only way to encode the depth information to imbue 3D-like characteristics is through simultaneous projection of phase mask and the depth information to the SLM. It is currently unknown if naively projecting them at the same time will work or additional measures need to be taken. Moreover, it is also important to figure out if we have successfully encoded the depth information without seeing the desired holography, because encoding the depth information is necessary but may not be sufficient for our final goal. These will be the top priorities to be addressed in the upcoming semester.

While we are currently utilizing the IFTA algorithm, it is more computationally intensive than the other algorithm Poon proposes in his book. The other algorithm accounts for the Spatial Distortion that occurs from Fresnel

Diffraction and thus premultiplies by the transfer function's inverse to cancel the aperture effects. We will try and get a working prototype with the IFTA algorithm first, but we can speed up computations with this other algorithm if it needed.

For the projection stage, as there is an apparent difference between the SLM reflective surface and a computer monitor, we hope to thoroughly test it the upcoming semester. While the automatically scaled image to fit the computer screen showed no issue, it should also be tested whether such adjustments potentially disturb the quality of holography. After confirmation of reliable projection, we hope to figure out how manipulating certain settings, e.g. the frequency and intensity of the laser beam, refresh rate of the screen, would incur qualitative changes to the projection.

References

[1] Maxim Shusteff, Allison E. M. Browar, Brett E. Kelly, Johannes Henriksson, Todd H. Weisgraber, Robert M. Panas, Nicholas X. Fang, and Christopher M. Spadaccini. One-step volumetric additive manufacturing of complex polymer structures. *Science Advances*, 3(12), 2017.

[2] Ting-Chung Poon and Jung-Ping Liu. *Introduction to Modern Digital Holography : With MATLAB*. Cambridge University Press, Cambridge, United Kingdom, 2014.

[3] Sebastien M. Popoff. wavefronthsaping/slmpy: First release, 2017. Accessed: 2018-11-30.