Exploiting Diffraction Orders from a Phase Limited SLM to Remove and Model Zernike Aberrations

by

Aaron J. Goschie

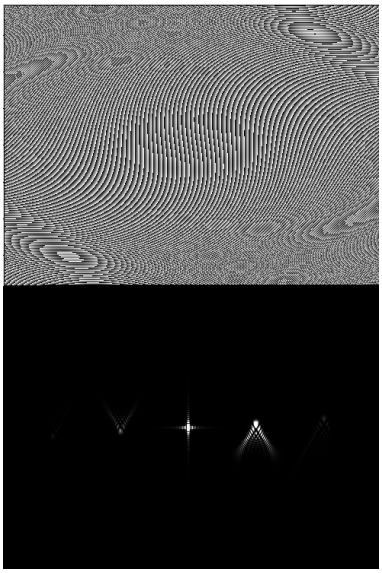
An undergraduate thesis advised by Professor David H. McIntyre

submitted to

the Department of Physics, Oregon State University

in partial fulfillment of the requirements for the degree of

BSc in Physics



May 10, 2019

Exploiting Diffraction Orders from a Phase Limited SLM to Remove and Model Zernike Aberrations

Aaron J. Goschie

Department of Physics, Oregon State University, Corvallis, Oregon

(Dated: May 27, 2019)

Abstract: Using a plastic bottle, we explore the capabilities of a reflective phase-only Spatial Light Modulator (SLM) in removing wavefront aberrations using a Zernike polynomial phase mask, with the modulators phase depth limited to 0.8π radians. Wavefront aberrations, which can be modeled with Zernike polynomials, distort image transmission through various optical devices, and there is interest in using the SLM for removing these phase-only aberrations in coherent monochromatic light (such as a laser). To measure the capabilities of the modulator mapped with the Zernike polynomials, we placed a plastic soda bottle in different positions of a Fourier optical systems beam path, such that we could measure the aberrational effects of the bottle on the point spread function. We found that by placing a reflective SLM with a retardation of 0.8π before the transform lens, pixel mapped with both a blazed grating and Zernike polynomials, we were thus capable of controlling the deviation and peak intensity of a particular diffraction order in the focal plane of the transform lens. By placing a CCD camera in the focal plane and imaging a non-zeroth diffraction order, we can measure the convoluted point spread function while ignoring the non-diffracting light at the center of the CCD. We find that 13 lower order Zernike polynomials are capable of reducing the deviation in the intensity of the first diffraction order while maximizing its peak brightness, but the result imposes certain conditions on the location and curvature of the soda bottle and may not have equal effects across all non-zero diffraction orders. From our data, we were able to conclude that Zernike polynomial phase masks, limited by an SLM with 0.8π retardation, is capable of removing aberrations in a laser beam when localized to a diffraction order, but "modeling" them was inconclusive.

CONTENTS

LIST OF FIGURES

I. INTRODUCTION

A. Motivation

Unwanted optical aberrations, in various systems, are a common effect observed in different scientific disciplines. The light we observe after it experiences an optical aberration can generally described as warped, blurred, or bent. In the field of astronomy, atmospherically generated aberrations are a consistent problem, as the Earth's atmosphere is a giant transparent medium, subject to statistical variations in its index of refraction [1]. By the time the light has passed through the Earth's atmosphere, through the telescope, and into the observers eye, the light will have already picked up various aberrations of different qualitative character.

In medical science, the accuracy of light based imaging systems is often limited by the sample medium itself, which produce aberrations in "wavefronts", such as in retinal imaging of the eye [2]. The cornea and fluid of the eye also represent a change in the index of refraction, and light rays passing through these media travel different optical path lengths, and so the light appears warped or bent. Understanding optical aberrations, and means of mitigating them with adaptive optical devices, is paramount to furthering any field which utilizes light-based imaging systems.

When aberrations build up in light waves, the regularity over the front of the traveling wave is disturbed. Optical imaging systems usually rely on this regularity to reproduce an accurate account of the object(s) in question. In astronomy, the light is so far away that rays begin to become a parallel wave front, and we are only able to perceive information up to a certain limit. But the Earth's atmosphere warps what little light an astronomer could collect along that wave front, and thus astronomical images are especially sensitive to aberrations.

An aberration does not necessarily make information unrecoverable, rather it alters the curvature of the wave front in a traveling light wave. Because we can theoretically recover information from an aberrated light wave, it is worth exploring adaptive optical devices and means of controlling them. First, we must define what a wave front aberration is.

B. A Wave Front With an Aberration

What we mean by the light's "wave front" is in the context of monochromatic electromagnetic waves, or for the added condition of coherence, a laser beam. A plane wave is a trivial example of such a class of light, and it will help highlight what an aberration across its "wave front" means in the context of this paper. We

will start with

$$\vec{E} = \vec{A_0} e^{i(\vec{k}\cdot\vec{r} - \omega t + \phi)}. \tag{1}$$

In equation (1), $\vec{E}$ is electric field of the plane wave. A_0 is the amplitude vector of this wave, with a wavelength $|\vec{k}|$ and angular frequency ω. The wave travels in the direction of $\vec{k}$. Due to these imposed parameters, the wave is best described as a set of regularly spaced planes with interval $\frac{2\pi}{|\vec{k}|}$, where the wave reaches its maximum amplitude $|\vec{A_0}|$ at each plane if we only consider the real part of the wave. These planes are the "wave front" which we discussed before. ϕ is its phase offset, and you could imagine it as shifting all the planes back and forth depending on its value. But the planes all have the same phase at the same distance in the direction $\vec{k}$, and so we do not consider the wave fronts to contain an aberration.

When the wave fronts do contain an aberration, ϕ becomes a more complicated parameter. The aberration creates spatial variability perpendicular to $\vec{k}$ at some axial position z_0', so we write:

$$\phi = \phi(x', y', z' = z_0'), \tag{2}$$

and we can regard x' and y' as the position in directions orthogonal to $\vec{k}$, and z' as some position in a direction parallel to $\vec{k}$. The wave fronts are still regularly spaced temporally, but the phase changes result in spatial shifts of the waves maxima. The maxima are no longer planes, and are now surfaces varying in 2-dimensions perpendicular to $\vec{k}$. Also, ϕ can be represented as a linear combination of component functions, which implies that aberrations are separable under the following condition:

$$\vec{E} = \vec{A_0} e^{i(\vec{k}\cdot\vec{r} - \omega t)} e^{i\phi_0} e^{i\phi_1} ... e^{i\phi_n}, \tag{3}$$

where ϕ_n is some separable aberration function of (x', y', z') with an implicit magnitude. Zernike polynomials have been found as a viable candidate for these separable aberrations. (Need citation)

C. Zernike Polynomials and Aberrations

In 1934, Fritz Zernike created a set of polynomials which were orthogonal on the unit disk [3]. More importantly, they could accurately curve fit the profile of common aberrations found across wave fronts. The polynomials adequately described the shape of wave fronts emitted as circular pupil planes in optical systems. In other words, in a system of circular lenses and mirrors, structurally curved distortions affect the wave front of light in the pupil planes, and the wave front is readily described by a linear combination of what Fritz Zernike called "Zernike polynomials". Because Zernike polynomials form a complete orthogonal

basis, they can represent image moments, and are known to effectively map an image on a disk as 2d basis functions, regardless of the images rotation angle. ϕ, from the wavefront described by equation (1), is an image of phase capable of being scaled to a circular pupil, and we are thus able to use Zernike polynomials as phase moments at some z_0 from equation (2); they will also be separable like in equation (3). Only a subset of the first 21 Zernike polynomials (Figure 1) are usually relevant in terms of modeling aberrations. They can represent tilt in x and y directions, defocus, astigmatisms, and other phase anomalies.

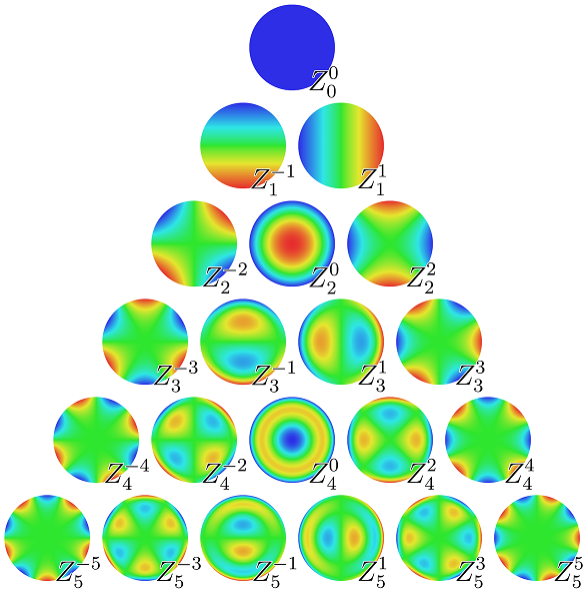


FIG. 1: Zernike Pyramid. The polynomials are counted from left to right, top to bottom, and represent the proportions of phase shift across a circular pupil (red being most intense). Figure extracted from [4].

Zernike polynomials are indexed according to n and m, which require that $-n \leq m \leq n$ and $n \geq 0$. We can ignore the polynomials with $n-m$ being odd, as those polynomials are equal to zero. As stated before, Zernike polynomials are orthogonal over the unit disk, thus the following is true over the unit disk:

$$\begin{aligned} \int Z_n^m(r,\varphi) Z_{n'}^{m'}(r,\varphi) dA &= \frac{\epsilon_m \pi}{2n+2} \delta_{n,n'} \delta_{m,m'}, \\ Z_n^{|m|} &= R_n^{|m|}(\rho) cos(|m|\Phi)), \\ Z_n^{-|m|} &= R_n^{|m|}(\rho) sin(|m|\Phi)), \\ R_n^{|m|}(\rho) &= \sum_{l=0}^{\frac{n-|m|}{2}} \frac{(-1)^l (n-l)!}{l!(\frac{n+|m|}{2}-l)!(\frac{n-|m|}{2}-l)!} \rho^{n-2l}, \end{aligned} \tag{4}$$

where Z_n^m is a Zernike polynomial and dA is a disk area element; equation (4) also assumes continuity. Obviously, our system is not continuous, thus we are obtaining an approximation for the magnitude of Zernike polynomials. If we define our variable phase from equation (2) to be $\phi(r,\varphi)$, and Zernike polynomials

form a complete orthogonal set, then we can represent ϕ as a summation of normalized Zernike polynomials with independent coefficients. Thus, the following is true assuming our wavefront has been scaled to the unit disk:

$$c_n^m = \frac{2n+2}{\epsilon_m \pi} \int \phi(r,\varphi) Z_n^m(r,\varphi) dA, \tag{5}$$

where c_n^m is the associated Zernike coefficient; it describes the magnitude of the Zernike polynomial contained in ϕ. If $m = 0$, $\epsilon_m = 2$ and 1 otherwise. $\phi(r,\varphi)$ must then be equivalent to

$$\phi(r,\varphi) = c_n^{-m|} Z_n^{-|m|} + ... + c_1^{-1} Z_1^{-1} + c_0^0 Z_0^0 + c_1^1 Z_1^1 + ... + c_n^{|m|} Z_n^{|m|}. \tag{6}$$

If we only use the (n, m) Zernike polynomials, then the coefficients represent the magnitude of the (n, m) associated aberrations, which form a linear combination like in equation (6). We only use 13 of the Zernike polynomials. If we re-index c_n^m to be c_k, then our 13 Zernike polynomials are $c_0 Z_0^0$, $c_1 Z_1^{-1}$,$c_2 Z_1^1$, $c_3 Z_2^0$, $c_4 Z_2^2$, $c_5 Z_2^{-2}$, $c_6 Z_3^{-1}$, $c_7 Z_3^1$, $c_8 Z_4^0$, $c_9 Z_3^{-3}$, $c_{10} Z_3^3$, $c_{11} Z_4^2$, and $c_{12} Z_4^{-2}$.

D. Objective

As initially stated, aberrations are not inherently restricted to lenses and mirrors; astronomical aberrations (aberrated light from the stars) are atmospherically generated. Zernike polynomials have also been successfully used to model atmospheric aberrations [1].

Because any transparent media is capable of being a distortion media, transparent solids which cause continuous variable phase shifts over a wavefront can, in general, be assumed as valid experimental subjects for quantification. The experiment outline in this paper will use a transparent plastic bottle as an aberrating medium at various positions in an optical apparatus. However, the practicality of quantification also depends on the adaptive optical device used to mitigate them.

Unlike the deformable mirrors used to remove aberrations in astronomical telescope systems [1], our optical bench based apparatus will use a reflective phase-only Spatial Light Modulator (SLM) of the "Liquid Crystal on Silicon" variety (LCOS). An SLM reflects light at its chip interface, which is formed as a contiguous rectangular grid of pixels. Each pixel is capable of inducing a localized phase shift by realigning liquid crystals in its vicinity (equivalent to increasing the optical path length), independent to the rest of the pixels. The experiment outlined in this paper will map Zernike polynomials to the pixel grid of the SLM in order to remove and model aberrations generated by the mentioned plastic bottle. The practicality of Zernike polynomials being an accurate "modeler" of aberrations, or as "removers" of aberrations, may depend on the maximum phase shift of these pixels. The hypothesis put forward is that an SLM with a limited maximum phase shift

is unable to remove or model aberrations using Zernike polynomials. Our experiment will use an SLM with a phase shift range of 0 to 0.8π radians to test this hypothesis, while a natural phase aberration ought to exist in the range of 0 to 2π radians.

On the contrary, the experimental results will later demonstrate that lower order Zernike polynomials are capable of removing aberrations generated by a plastic bottle if they are mapped to an SLM with a maximum phase of 0.8π, but the system may not always accurately "model" the aberrations.

In summary, the objective is to use an SLM of 0.8π maximum phase (mapped with Zernike polynomials) to modulate the phase of a laser beams wavefront, such that it will correct (or not correct) the wavefront after it has passed through the plastic bottle, or before it passes through the plastic bottle (at different positions), and consequently reveal if it can form the unaberrated image at the end of the beam path. The same SLM will be mapped with a blazed grating to produce a diffraction pattern (the intended image) in the focal plane of the lens. We hypothesized that, if the Zernike polynomials create a phase shift which cancels with aberrations of the same form as equation (3), an intentionally modulated laser beam of a similar form can create a sharper diffraction pattern in the focal plane of a lens, when compared to a non-modulated beam. Our objective also entails measuring the sharpness of the diffraction pattern in real time, utilizing variable coefficients for each Zernike polynomial to change their magnitude in an iterative fashion. The theory of the system relies on the properties of a collimated Gaussian beam reflecting off of a blazed grating, and propagating through a lens into the focal plane. The theory can be further explained in terms of Fresnel diffraction, Fourier optics, and the effects of Zernike aberrations on a PSF (point spread function). These concepts will be outlined in more detail in the theory section.

II. THEORY

A. Blazed Grating and a Lens

The SLM's pixel grid could be considered a mirror-like aperture. Assume that f is the focal length of a lens and the aperture to lens distance is $d = f$, and also assume the Fresnel limit holds for light traversing distance d to the lens. When light reflects off of the aperture and passes through the lens to the other side, the lights electric field amplitude undergoes a Fourier transform when observed across the focal plane [5]. If we make sure the lens aperture is wider than the SLM aperture, the electric field amplitude measured in the focal plane is nearly equivalent to Fraunhofer diffraction, where we measure light coming from an aperture that is extremely far away, and there is approximately no phase curvature. Thus, we obtain:

$$\begin{aligned} E_{focal}(x,y) &= -\frac{ie^{2ikf}}{\lambda f}\int\int E_{aperture}(x',y')e^{-i\frac{k}{f}(x'x+y'y)}dx'dy', \\ &\propto F\{E_{aperture}(x',y')\}(x,y), \end{aligned} \tag{7}$$

where f is the focal length of the lens, λ is the wavelength, and $k = \frac{2\pi}{\lambda}$. The definite integral of equation (7) covers the entire aperture plane spanned by (x', y'), while the entire focal plane is described by (x, y), and is thus proportional to a 2D Fourier Transform $F\{E_{aperture}(x', y')\}(x, y)$. Now assume the aperture function $E_{aperture}$ is equivalent to

$$E_{aperture} = A(x',y')e^{i\phi_{blaze}(x')}. \tag{8}$$

$A(x', y')$ is the amplitude of the aperture function, and $\phi_{blaze}(x')$ is a periodic sawtooth phase profile, which only varies in the x' direction. $\phi_{blaze}(x')$ is physically proportional to a blazed grating, and according to Hecht on pages 496-502, reflects light from a sawtooth shaped periodic optical path-length difference like seen in Figure 2 [6]. The reflected rays only constructively interfere at certain spots, which are called diffraction orders. Later, we will discuss how the SLM has non-diffracted light in the zeroth diffraction order, and why we require a periodic phase due to that effect. If we assume a light wave reflects off of the aperture, with the aperture function of equation (8) set to $A(x', y') = 1$, then the diffraction orders observed at $E_{focal}(x, y)$ will be exactly equivalent to Figure 2, assuming diffraction angles β that are smaller than the angular width of the lens from the central point of the aperture. Thus, the transform of equation (8) has diffraction orders like the grating equation

$$d(sin(\alpha) - sin(\beta)) = m\lambda. \tag{9}$$

The parameters are described in Figure 2, with diffraction orders being denoted by integers m. An infinite grating would result in diffraction orders corresponding to Dirac delta functions in the Fraunhofer limit [6]. Considering equation (7), this would imply that $F\{e^{i\phi_{blaze}(x')}\}(x,y) = Envelope(x,y)comb(x,y)$. The

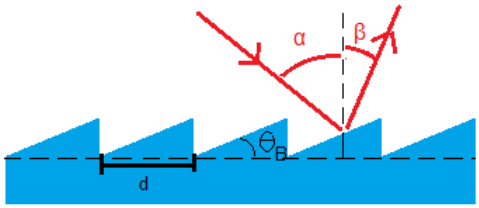


FIG. 2: Blazed Grating. Blazed grating with slope angle θ_B and scale d. Light incident at angle α dictates the angle of diffraction orders at angle β. Light rays will constructively interfere at a non-zero diffraction order with the greatest intensity.

$comb(x,y)$ is a repeating train of Dirac delta functions. If we make $A(x',y')$ the window function such that $A(x',y') = 1$ only for $-\frac{W}{2} < x' < \frac{W}{2}$ and $-\frac{L}{2} < y' < \frac{L}{2}$, ($W$ and L are the length and width of the window) and we apply the convolution theorem as described on pages 557-565 of Hecht [6], we find that equation (7) is proportional to

$$E_{focal} \propto F\{A\}(x,y) * Envelope(x,y)Comb(x,y). \tag{10}$$

The $*$ is the convolution operator and equation (10) must reduce down to a summation of scaled copies of $F\{A\}(x-a_m,y)$, where a_m is the x-axis position of each diffraction order. As long as the periodicity of the diffraction grating is smaller than the dimensions of the aperture and lens, the function $F\{A\}(x-a_m,y)$ will not overlap with other diffraction orders, as the $A(x',y')$ in our system is around $10mm$ wide and the lens focal length is around $200mm$. As seen in Figure 3, the observable intensity of $F\{A\}(x,y)$ is narrow and well peaked. $F\{A\}(x,y)$ in this circumstance is known as the PSF (Point Spread Function), as it represents the spread of a light point source upon image formation through an optical system.

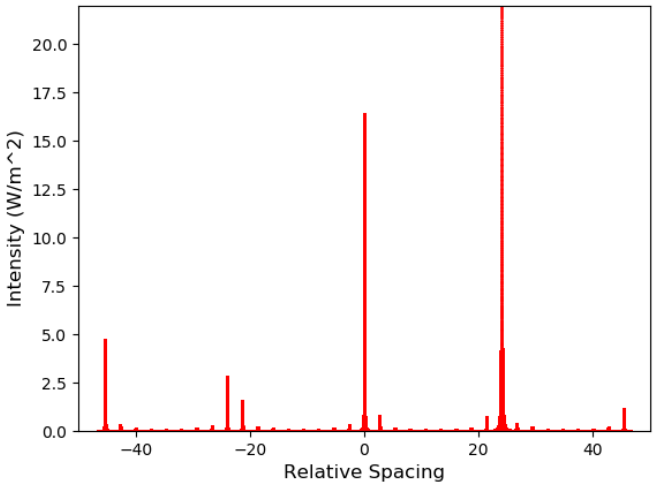


FIG. 3: Blazed Diffraction Using a Finite Aperture. A 1D simulation of a blazed grating (Appendix A: SLMSimulator1D.py) runs with a $10mm$ wide aperture and ramp spacing of $d = 50\mu m$, with a θ_b such that 1.4π max phase shift occurs; the lens is $200mm$ and the intensity is from a collimated $630nm$ laser. From equation (10), each spike is the observed intensity of a diffraction order in $Comb(x, y)$ (centered at $y = 0$), with some peaking at a different intensity due to $Envelope(x, y)$. The $+1$ diffraction order is the tallest peak to the right of zeroth order central peak.

B. Effect of a Collimated Gaussian Laser

In the Fraunhofer limit, the diffraction pattern from a blazed grating will appear as discrete spots of light called diffraction orders. We will be measuring the intensity distributions of a diffraction orders PSF, and the type of laser beam used with the blazed grating will affect the PSF. A laser beam with spatially Gaussian convergence will ensure localized measurements. A linearly polarized Gaussian laser beam is a solution to Maxwell derived wave equations with a Gaussian radial intensity profile and azimuthal symmetry [7]. Most researchers refer to the electric field component in the direction of polarization, which is defined as the following:

$$E_x = \frac{E_0 e^{\frac{-r^2}{w^2(z)}} g(\varphi)}{\sqrt{1 + \frac{z^2}{z_0^2}}} e^{i(kz[1+r^2/2(z^2+z_0^2)]-\omega t - tan^{-1}(z/z_0)}, \tag{11}$$

where r and z are the cylindrical coordinates, with z being the propagation position. We can assume $g(\varphi)$ is constant, as we are assuming the laser is not temporally pulsed. The symmetry in Equation (7) is the only important factor for an accurate metric of a diffraction order, where the symmetric Gaussian is $e^{\frac{-r^2}{w^2(z)}}$. In a

collimated Gaussian beam, the amplitude factors are approximately constant over the propagation distance, so we write:

$$\frac{E_0 e^{\frac{-r^2}{w^2(z)}}}{\sqrt{1+\frac{z^2}{z_0^2}}} \approx E_0' e^{\frac{-r^2}{|\gamma|}}, \tag{12}$$

where $|\gamma|$ is the width parameter of the Gaussian. Using the constant Gaussian profile of equation (12), the convolution of a diffraction order with the PSF of a Gaussian beam should also approximately be Gaussian when multiplying it into equation (7) and consequently convolving into equation (10), forming

$$E_{focal} \propto F\{A\}(x,y) * F\{E_0' e^{\frac{-x'^2-y'^2}{|\gamma|}}\}(x,y) * Envelope(x,y)Comb(x,y). \tag{13}$$

Equation (13) is the focal electric field, which is equivalent to a Gaussian laser beam reflected off of a blazed grating, and then Fraunhofer transformed through a lens. Thus, with the correct constant parameter $|\gamma|$, any measurement of the diffraction orders intensity distribution should behave as if its locally convolved with a Gaussian distribution, with a near rotationally symmetric deviation and peak intensity in the best case.

C. Effect of The SLM As a Zernike Aberration Modulator

A reflective SLM is a computer connected device containing an array of liquid crystal pixel elements. We are using a TN (twisted nematic) type of spatial light modulator, and so there is a polarization effect if the polarization of a reflecting light source is not linearly aligned properly. If the incident light source is linearly polarized to the proper alignment, like in our case, the SLM becomes a phase-only spatial light modulator. Our SLM has 1024x768 pixels, each with an area of $9\mu m$ x $9\mu m$, and each pixel element can be controlled by sending an 8 bit gray scale value to the device, along with the pixels address. As seen in Figure 4, each pixel can be assigned a value from 0-255, the value corresponding to the amount of phase shift the incident light will experience upon being reflected. Using our SLM, the 0-255 value range (approximately) linearly maps to a phase shift between 0-0.8π. Due to the discrete nature of the device, the efficiency of light modulation comes with a certain level of error which has been described by previous researchers with David H. McIntyre [8]. Also, because the diffraction efficiency is limited in our SLM, the zeroth diffraction order is flooded with non-diffracted light, which is light experiencing the SLM as a simple mirror. The non-diffracted light is the reason for mapping a periodic phase to the SLM, such that we can measure the intensity distribution of a separate diffraction orders PSF without pollution.

The individual degree of control is highly predictable, and can be fully automated using LabView software or other interface methods. Due to a liquid crystal pixel grid being analogous to an LCD screen,

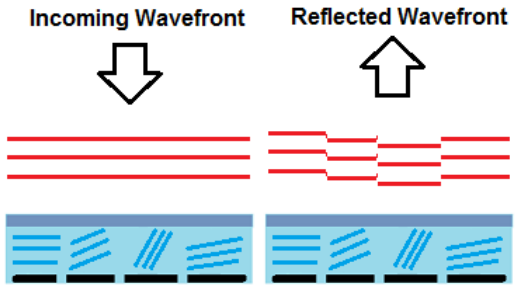


FIG. 4: SLM Interface. Light waves reflecting off the liquid crystal interface of an SLM (left and right). Each of the 4 pixels at the bottom of the image stimulates the liquid crystals to phase shift the reflecting wavefront, as seen on the right of the image.

we can reuse LCD screen interfaces to communicate with the device, and additionally view the gray scale phase map on a computer monitor.

We are sending Zernike polynomials as phase data to the SLM and exploring its effect in canceling out aberrations from natural media. To map the Zernike polynomials, they must be scaled to map to the rectangular pixel grid of the SLM. The true radius of the Zernike polynomial must then be the shortest axis of the pixel grid divided by two, which is $R = 3456\mu m$, if we assume our dimensions are in micrometers. For example, the Zernike polynomial proportional to Z_2^0 (called defocus) is scaled by dividing the cylindrical coordinate radius r by R:

$$c_3 Z_2^0(\frac{r}{R}) = ZP_{defocus}(r) = c_{defocus}(2(\frac{r}{R})^2 - 1), \tag{14}$$

where we assign some magnitude $c_{defocus}$ to the original re-indexed coefficient c_3. Next, the Zernike polynomial must be mapped to the SLM, which constitutes a discrete representation. We then convert defocus into its discrete Cartesian form:

$$ZP_{defocus}(i', j') = modulo(c_{defocus}(\frac{(i+0.5)'^2 81\mu m^2}{R^2} + \frac{(j+0.5)'^2 81\mu m^2}{R^2} - 1), 256), \tag{15}$$

where i' and j' are integers for the pixel indices corresponding to the x-axis and y-axis, and centered on the SLM's pixel grid such that $-512 <= i' < 512$ and $-384 <= j' < 384$. **We assume that the parts of $ZP_{defocus}$ that are not within the radius R will have negligible effects.** The polynomial undergoes modulo arithemetic up to 256, as there is a bit range from 0-255. Values that cycle over 255 repeat and represent cycling through radians of phase. As stated before, the SLM we are using is 0.8π limited in phase,

so after 255, the phase drops from 0.8π to 0.

Now lets assume the SLM is also discretely mapped with $\phi_{blaze}(i')$ from equation (8) and that it to can only reach a maximum of 0.8π phase. If we compile equations (7), (8), and (15), our new relationship between E_{focal} and $E_{aperture}$ is

$$E_{focal}(x,y) = -\frac{ie^{2ikf}}{\lambda f}\int\int A(x',y')e^{i\cdot modulo(\phi_{blaze}(i')+0.8\pi\frac{ZP_{defocus}(i',j')}{255},0.8\pi)}e^{-i\frac{k}{f}(x'x+y'y)}dx'dy', \quad (16)$$

where x' and y' are in micrometers, and assign $i' = floor(\frac{x'}{9\mu m})$ and $j' = floor(\frac{y'}{9\mu m})$. Equation (16) represents the pixel phases mapped to the SLM as a phasor that can only cycle from 0 to 0.8π. We then see that E_{focal} is proportional to the Fourier transform of the product between the modulo 0.8π phasor and aperture function. Lets also introduce the collimated Gaussian profile of our incident beam in the Fourier transform, which results in

$$E_{focal} \propto F\{A(x',y')e^{\frac{-x'^2-y'^2}{|\gamma|}}e^{i\cdot modulo(\phi_{blaze}(i')+0.8\pi\frac{ZP_{defocus}(i',j')}{255},0.8\pi)}\}. \quad (17)$$

Now, assume we could replace $ZP_{defocus}$ with any Zernike polynomial combination ZP_{any}, similar to equation (6), but still limited to modulo 256. The SLM would be capable of mapping any combination of Zernike aberrations, but there would be a phase limitation. There will still be an interaction between the SLM phase and natural aberrations, but they could not match each other. Also, lets assume those natural aberrations exist in our system and form a combination of Zernike polynomials like equation (6), but scaled such that the shortest axis of the SLM's rectangular pixel grid is their radius. They would be separable like from equation (3), but lets describe all of them as the same function $\phi_{2\pi}(x',y')$, which does cycle through a full 2π phase. And finally, if we simplify E_{focal} using the convolution theorem as we have done before to obtain

$$E_{focal}(x,y) \propto F\{A(x',y')\} * F\{e^{\frac{-x'^2-y'^2}{|\gamma|}}\} * F\{e^{i\cdot modulo(\phi_{blaze}(i')+0.8\pi\frac{ZP_{any}(i',j')}{255},0.8\pi)}\} * F\{e^{i\phi_{2\pi}(x',y')}\}. \quad (18)$$

In equation (18), if there was no modulo operation involved, then Zernike polynomials in ZP_{any} of equal but opposite magnitude to polynomials in $\phi_{2\pi}(x',y')$ would approximately cancel out (assuming our pixels are small), leaving just the $\phi_{blaze}(i')$. The result would be equivalent to removing the natural Zernike aberrations. We would see the diffraction orders in E_{focal} reduce to their original PSF, like in equation (13). But the modulo function makes ZP_{any} inseparable from $\phi_{blaze}(i')$, so the removal of aberrations would have to depend on the effect of $F\{e^{imodulo(\phi_{blaze}(i')+0.8\pi\frac{ZP_{any}(i',j')}{255},0.8\pi)}\}$ on the PSF, assuming $Comb(x,y)$ is still generated like in equations (9) and (10), and ZP_{any} results in a localized PSF.

From equation (18), we can see that $F\{A(x',y')\}$, $F\{e^{\frac{-x'^2-y'^2}{|\gamma|}}\}$, and $F\{e^{i\phi_{2\pi}(x',y')}\}$ would still be convoluted normally with each diffraction order under the assumption that $Comb(x,y)$ is still generated. If

ZP_{any} in any way reduces aberrations from $\phi_{2\pi}(x', y')$, then we would mostly recover the original PSF and we could measure that change on a diffraction order. The change would be an output, while ZP_{any} would be an experimental parameter. According to James C. Wyant and Katherine Creath, the PSF of diffraction orders become less peaked in intensity due to Zernike phase aberrations, as there is an effect on the Strehl ratio [3]. Thus, to determine the removal of Zernike aberrations from a diffraction orders PSF, we measure the peak intensity and deviation in intensity. To maximize one and minimize the other would indicate less Zernike aberrations. If the Zernike aberrations were perfectly removed, every diffraction order would see the same maximization of intensity and minimization in deviation. The theory does not seem to agree, that a global improvement resulting in the original PSF of each diffraction order, is possible.

III. METHODS

Figure 5 is the optical layout of the experiment. By the time the red laser beam reaches the CCD camera, it will take the form of a Fraunhofer diffraction pattern due to L2. And as described in equation (7), the entire optical system results in a 2d Fourier transform of the accumulated electric field just after the SLM, but in the plane of the CCD. In our apparatus, L2 is a converging lens with a focal length of approximately

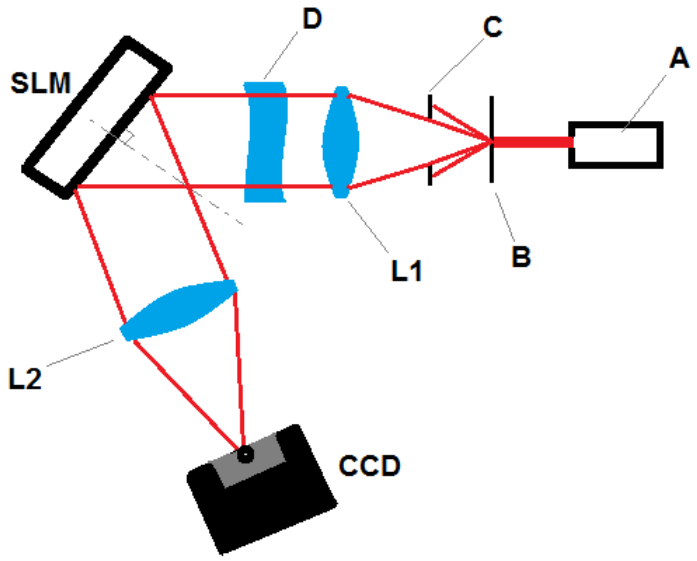


FIG. 5: Experiment Diagram. A red laser beam (A) is spatially cleaned by a pinhole aperture (B) and iris (C), which only allows an airy disk pattern to illuminate the first converging lens (L1). The beam is now parallel and picks up aberrations through a transparent medium (D). A reflective SLM modulates the beam at its chip interface with a blazed grating, and finally reflects the beam through the second converging lens (L2). A Fraunhofer diffraction pattern forms on the CCD.

$200mm$. The distance between the SLM and L2, and between L2 to the CCD, is equal to the focal length to make the Fraunhofer transform in equation (7) equivalent. As the beam propagates through the medium (D), there are variations in the optical path length over the cross section of the beam, which results in phase shifts of similar form, and equivalently phase aberrations. The beam then reflects off of the SLM, which needs to modulate the phase of the beam with, at minimum, a grating pattern (in this case a blazed grating). Diffraction orders are generated by the blazed grating, and recall that they are necessary for equation (18) to be valid, which allows us to measure the effects of the aberrations. A non-zero diffraction order, which does not exceed the intensity range of the CCD camera, is measured using a "spot metric", while discrete Zernike polynomial phase masks are applied to the SLM based on the pixel indices. They are then added to the blazed phase mask (with a blazed phase profile like Figure 2) using modulo addition, which results in an electric field at the CCD similar to Fraunhofer transform of equation (18).

A. The Spot Metric

We are collecting our physical data using a spot sharpness metric, or simply called the "spot metric". The spot metric is useful for monitoring the spread of light in computer connected cameras. More specifically, we are using it to describe how well a laser spot is "focused", particularly in the context of Gaussian lasers and the 1st order diffracted beam from our blazed grating. It should be strongly noted that the spot metric is **unitless**, and that it reaches a minimum when the peak intensity is highest and the deviation in intensity is lowest, which was demanded for measuring a Zernike aberrated PSF from section (II, C). We approximate the spot metric with a spread function equivalent to

$$S_{metric} = \frac{(\sum_{ij}^{N} I_{ij})^2}{\sum_{ij}^{N} I_{ij}^2}. \tag{19}$$

In equation (19), I_{ij} is a single pixels intensity value in our intensity matrix I. The pixels take up positions in the i row and j column, with the indices also corresponding to a row and column in the CCD chips pixel grid. As a result, each pixel corresponds to a location on the CCD camera, which measures a single light intensity value using 8 bits (0-255). We call the intensity matrix's location on the CCD camera the ROI (region of interest), which is selected in a manner similar to the simulation of Figure 6. The intensity matrix thus represents a subsection of the camera pixel grid receiving an arbitrary light source.

We only collected pixel data from a single diffraction order on the CCD camera. The +1 order diffraction spot was used as our data source. Its pixels can be isolated as its own intensity matrix by only considering the non-zero pixels associated with it, and then the spot metric can be applied to its collection of elements; this is how we measured the aberrations which changed the focus of a single diffraction order.

In series, we iterated through each of our 13 discrete Zernike polynomials by applying their phask masks one at a time to the SLM. While the same Zernike polynomial was addressed to the SLM, we incremented its magnitude coefficient while recording the spot metric of the diffraction order. For example, if we were currently measuring the effect of defocus, we would be incrementing the coefficient in equation (15) as we repeatedly updated the SLM's phase mask. After a Zernike polynomial is done incrementing its coefficient, the coefficient associated with the most minimum spot metric is applied permanently, and the iteration continues to the next Zernike polynomial. Eventually, a linear combination of our 13 Zernike polynomials (in discrete form) is the result of the iteration, forming the complete ZP_{any} in the Fraunhofer transform from equation (18). The total phase looks similar to added up Zernike polynomials in equation (6), but each Zernike polynomial is discrete like equation (15). The resulting Zernike polynomial phase mask and coefficients will correspond to the lowest possible spot metric (S_{metric}) the polynomials are able to produce.

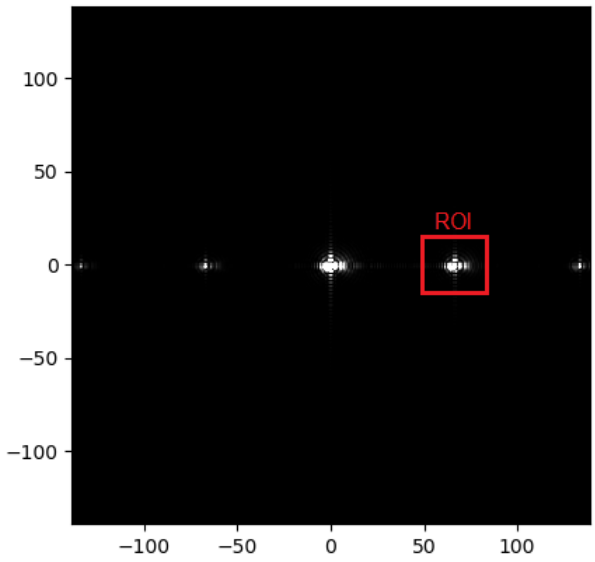


FIG. 6: ROI On Simulated Diffraction Pattern. The ROI area is chosen like the red ROI box, located over the +1 Diffraction order of the simulated Fraunhofer Diffraction pattern (Appendix B: SLMSimulator2D.py). The ROI in our experiment is selected from a live CCD camera using LabView automation software, which records a pattern like the simulation.

The data let us build graphs of the spot metric changing as we changed the coefficient for each Zernike polynomial, and these methods should result in a series of parabolic curves, where the spot metric is a function of a Zernike polynomials coefficient linearly changing through time (see Results and Discussion).

B. Eliminating Environmental Aberrations

In order to separate natural environmental aberrations from our distorting medium (D, Figure 5), we needed to find aberrations in the beam path before adding the distorting medium itself. In summary, we applied the series iteration method with step D in Figure 5 removed, yielding a Zernike polynomial phase mask and the associated coefficients. Leaving the coefficients the same, we could place the distorting medium at step D, and we applied the series iteration method with the control aberrations already minimized. But there were several issues that must be addressed. The first is the tilt aberration, which will move the control image.

a. Problems with tilt In order to measure the Zernike polynomials, we must apply and increment each Zernike polynomial independently and select the coefficient which yields the lowest spot metric. The problem is the Z_1^{-1} and Z_1^1 Zernike polynomials will move our control image out of the ROI that we are converting

to an intensity matrix. So every time we attempt to measure the validity of a new coefficient in reference to equation (19), the control image has been moved, and possibly out of the ROI, which would give a false measurement. Prior researchers working with David H. McIntyre have disregarded these particular polynomials in automated systems, and instead calibrated for them manually, and we are doing the same here.

b. The Undefined Path The path between the SLM and the camera is not able to modulated in the same way that the incident beam is able to be modulated. The path between the Camera and the SLM must be calibrated manually as well and not disrupted between experimental trials.

c. Control Transparent glass found in the lenses and dampeners of our apparatus form the primary medium that we will be experimenting with, and most often represents part of the medium of other experimental apparatuses (for instance microbiology). Glass is generally accepted to have an index of refraction of 1.513. We manually adjusted the Zernike polynomials associated with tilt, and recorded the spot metric as we varied the other Zernike coefficients, using Labview as an automated process. We found that tilt was ignorable, and we did not use tilt in our system. Perfect glass will simply refract the laser light that passes through it and bend the beam slightly. But we are measuring imperfections using Zernike polynomials. Glass can be slightly warped and lensed, and even clouded such that the index of refraction has substantially changed, which may impart Zernike polynomials on a transmitted wavefront.

d. Transparent Soda Bottle The distorting medium we used for step D in Figure 5 was an empty soda bottle. A soda bottle is made out of transparent bendable plastics. The refraction index of plastic is comparable to glass and on the order of $n \approx 1.5$. We chose a plastic soda bottle with a $20cm$ circumference and $10cm$ of usable height in the smooth region. The geometry of the plastic bottle (curvature), and micrometer variations in the plastic thickness, will result in continuous changes of the optical path length across the surface of a transmitted wavefront. Zernike aberrations should exist in the wavefront after experiencing the soda bottles medium, but they may or may not be eliminated from diffraction orders by modulating the SLM. Thus, we can apply the same process as part C to detect their elimination or lack thereof.

The SLM has a finite in size LCD chip, which acts as a mirror. The lens L2 in the Figure 5 is also of finite size. The finite size of these components limits the scattering information which reaches the CCD camera. Large angle scattering bypasses the SLM LCD chip or L2. This results in a loss of information which could have been measured in the field of the CCD. However, only the diffraction orders in the CCD can be measured, which may lack useful scattering information only available outside the chip area of the CCD. These are information trade-offs associated with finite apertures. Consequentially, the measurements at the CCD (in the ROI) should avoid scattering pollution to provide consistency. Physically modifying the experiment while remaining within the parameters of Figure 5 should be an adequate way to reduce the pollution in the ROI, such as tilting dampeners to prevent internal reflections.

IV. RESULTS AND DISCUSSION

A. Soda Bottle Results

The control was taken using the series iteration method in section (III, A), while taking the average of 3 spot metrics per increment to increase the accuracy. The beginning spot metric was $S_{control\_start} = 38.5$, and the ending spot metric was $S_{control\_end} = 22.0$. The change in spot metric of the control was $\Delta S_{control} = -16.5$, which was an improvement of $\%S_{control} = 42.9\%$ from $S_{control\_start}$. The improvement calculations are done using both

$$\begin{aligned} \Delta S &= S_{end} - S_{start} \\ \%S &= 100\frac{-\Delta S}{S_{start}}, \end{aligned} \tag{20}$$

where ΔS is the change in spot metric from beginning to end, and $\%S$ is the percent improvement in the spot metric; it can be thought of as the **diffraction focusing efficiency**. After obtaining the linear combination of Zernike polynomials for the control phase mask, we kept their coefficients the same and made them the preset phase mask for the soda bottle experiment. We physically placed the soda bottle between lens L1 and the SLM, like D in Figure 5. With the soda bottle in place, we used the series iteration method described in section (III, A) to find the linear combination of Zernike polynomials, but now with the effect of aberrations from the bottle. In Figure 7, we compiled the data into a graph of the +1 diffraction orders spot metric which we will call S_{L1_SLM}. 1 second corresponds to a single increment of a Zernike polynomial coefficient. We recorded parabolic minimization of S_{L1_SLM} over incremental variations in their magnitude. In Figure 7, the initial spot metric was $S_{L1\_SLM\_start} = 59.5$, while the ending spot metric was $S_{L1\_SLM\_end} = 34.4$. The improvement was $\Delta S_{L1\_SLM} = -25.1$ at $\%S_{L1\_SLM} = 42.2\%$. $\Delta R_{L1\_SLM} = 12.4$ was the difference between the $S_{control\_end}$ and $S_{L1\_SLM\_end}$, and represents the spot metric not removed from the iteration method, while $\%\Delta R_{L1\_SLM} = 33.1\%$ is the percent of spot metric not reduced between $S_{L1\_SLM\_start}$ and $S_{control\_end}$. A $\%\Delta R$ value of 0% would indicate a return to control's diffraction focus, and $S_{L1\_SLM\_end} = S_{control\_end}$.

The 5th and possibly 6th parabola in Figure 7 have a double bottom, which means two separate coefficients were plausible.

We also moved the soda bottle to a position between the IRIS and L1 from Figure 5. Again, we used the series iteration method on the +1 diffraction order, but we we had to expand the ROI from 10x10 pixels to 15x15 pixels due to a large PSF. $\Delta S_{IRIS\_L1\_start} = 116.0$, while $S_{IRIS\_L1\_end} = 88.3$. $\Delta S_{IRIS\_L1} = -27.7$ while $\%S_{IRIS\_L1} = 23.9\%$. $\Delta R_{IRIS\_L1} = 66.3$ and $\%\Delta R_{IRIS\_L1} = 70.5\%$

In Figure 8, we combined the the spot metric data from the control position, L1 to SLM position, and IRIS to L1 position, which details the $\%\Delta S$ trend of each position and recorded coefficient values c_k. Where

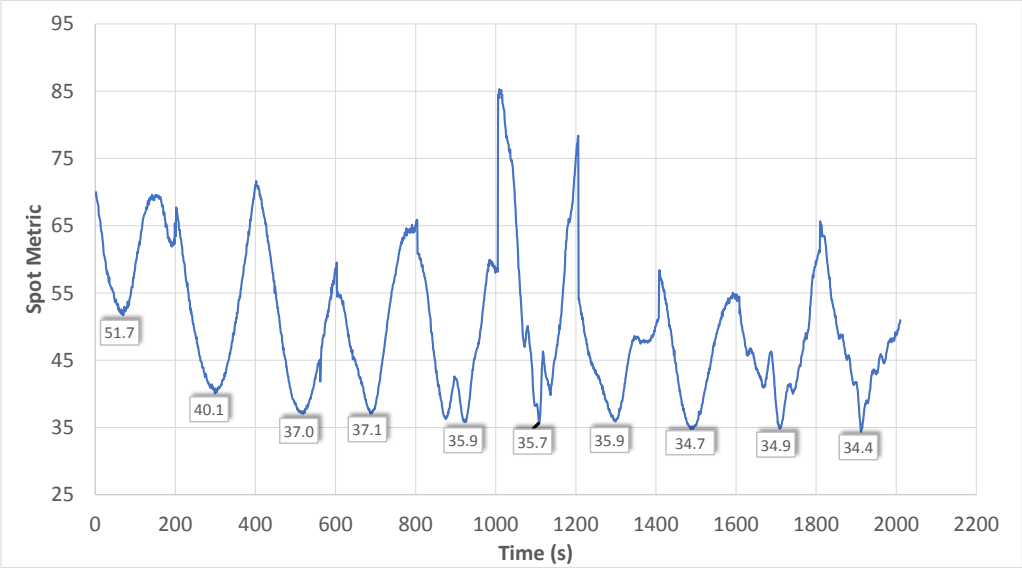
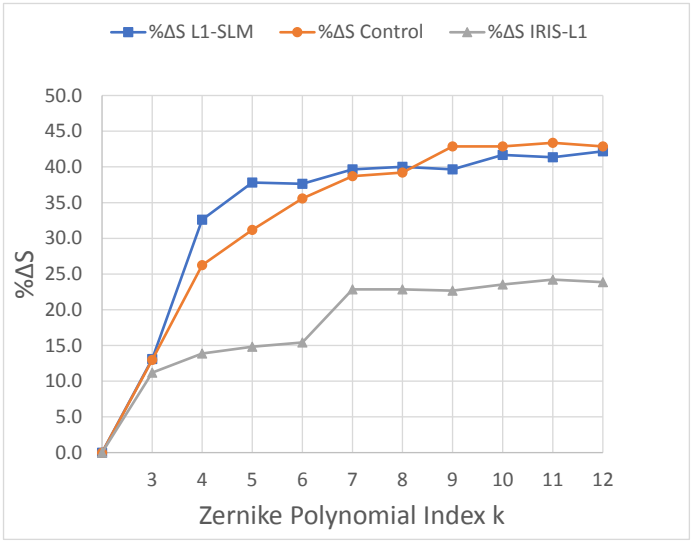


FIG. 7: Minimizing Soda Bottle Between L1 and SLM. The series iteration method from section (III, A) runs while a soda bottle was placed between L1 and the SLM. After approximately 2000 seconds, the iteration method completes. Each parabola is a sweep through the coefficient of one Zernike polynomial. Only 9 sweeps occur because the first 3 polynomials of our 13 Zernike polynomials are ignored. The parabolas sweep through the polynomial coefficients c_k, in the order of c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, and c_{12}, which correspond to the polynomials described at the end of section (I, C). The minimum spot metric can be seen under each parabola, marking the point when the coefficient in the sweep was selected.

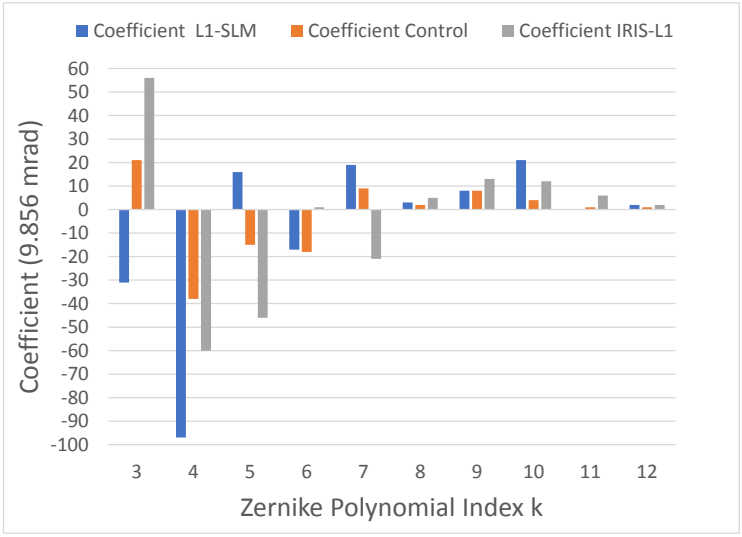
k is the index of the 13 Zernike polynomials we are using, and is stated at the end of section (I, C).

As can be seen in Figure 8 (a), the diffraction focusing efficiency improves after each optimal coefficient is added, eventually reaching a maximum. The SLM, mapped with the Zernike polynomials, has a similar diffraction focusing efficiency between the control and when the soda bottle is placed between L1 and the SLM. However, when the soda bottle is placed between the IRIS and L1, the diffraction focusing efficiency maximizes at a lower percentage. The drop in efficiency could be due to a disruption in the collimation of the beam itself, which disturbs the stability of any Zernike polynomials on the wavefront.

The ROI was expanded to 30x30 pixels for a before and after picture of the +1 diffraction order when the bottle was between L1 and the SLM, and the -1 diffraction order when the bottle was between the IRIS and L1. The -1 diffraction order was optimized separately from the data set taken in Figure 8. Figure 9



(a)



(b)

FIG. 8: Diffraction Focusing Efficiency. The diffraction focusing efficiency as a function of the Zernike polynomial index k (a). The optimal coefficient value that was found for each Zernike polynomial index k (b). Starting at $k = 3$, the series iteration method finds the coefficient that minimizes the spot metric (diffraction focusing efficiency maximizes), and permanently applies them one at at time, from left to right. For example, the largest coefficient was $c_3 \approx -97$.

shows qualitative improvement in the peak intensity of the series iteration for the L1 to SLM position, but only a marginal improvement during a series iteration of the IRIS to SLM position. Due to the increase

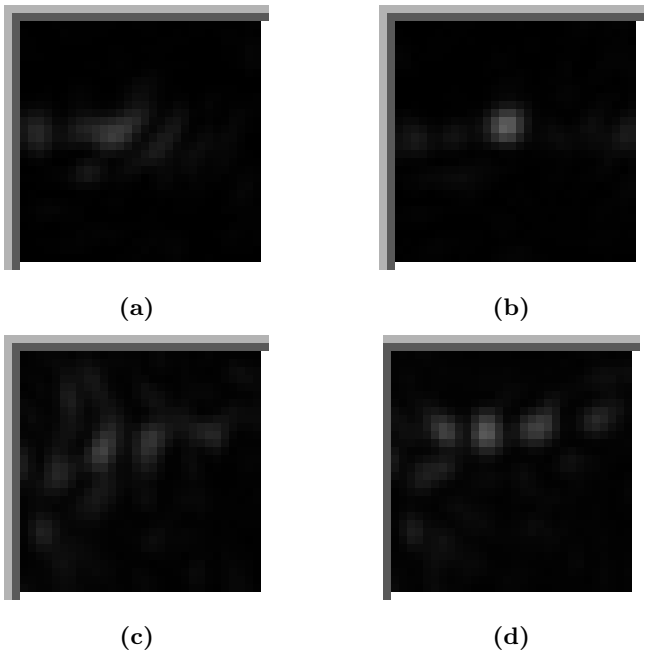


FIG. 9: ROI of Soda Bottle Positions. The ROI of the +1 diffraction order before the iteration method, with the bottle between L1 and SLM (a). The ROI of the +1 diffraction order after the iteration method, with the bottle between L1 and SLM (b). The ROI of the -1 diffraction order before the iteration method, with the bottle between IRIS and L1 (c). The ROI of the -1 diffraction order after the iteration method, with the bottle between IRIS and L1 (d). The -1 diffraction order, for the bottle between the IRIS and L1, was focused by the iteration method, but was only marginally improved, similarly to the +1 diffraction order in the Figure 8 data set. The ROI is 30x30 pixels in each image.

in diffraction focusing efficiency at each bottle position, and from the qualitative improvement in the ROI observed in Figure 9, a 0.8π phase limited SLM is able to remove aberrations from a plastic bottle. If the bottle is placed between L1 and the SLM, the diffraction focusing efficiency of the system is approximately 40%, which is similar to the control. Thus, the bottle creates aberrations in a manner similar to the glass optics of the rest of the system.

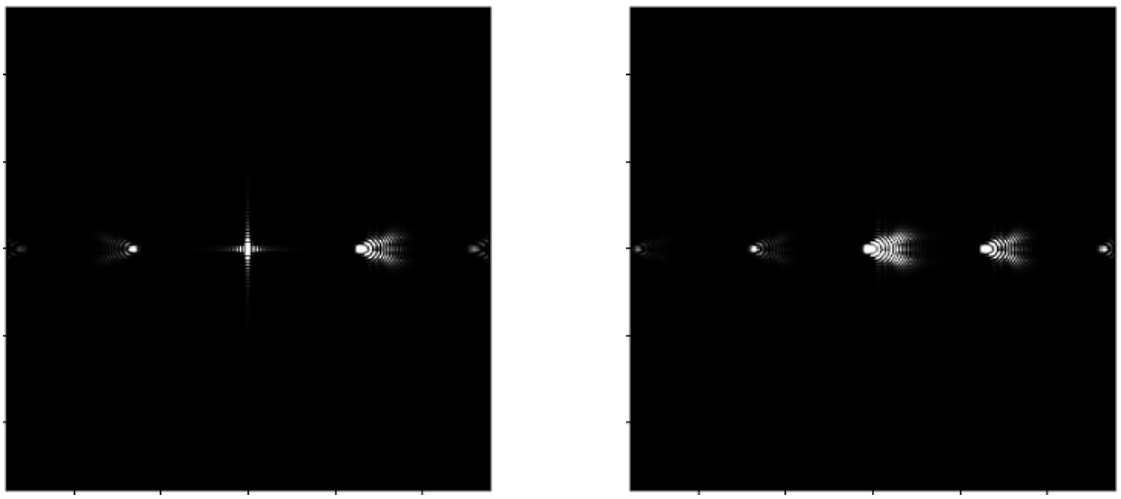
If the plastic soda bottle is placed between the IRIS and the collimating lens L1, the diffraction focusing efficiency of the system is reduced by approximately $\frac{1}{2}$, which implies that an initially parallel wavefront is required for optimal aberration removal.

The parabolic curves from Figure 7 demonstrate that each Polynomial could independently minimize the spot metric, which means that when only observing the +1 diffraction order, some orthogonality is maintained between interacting Zernike polynomial phase masks, despite the 0.8π phase limitation of the

SLM. However, whether or not the SLM mapped Zernike polynomial coefficients **model** the actual natural Zernike aberration coefficients, is not conclusively found in the data set.

B. Simulation and Different Diffraction Orders

A Fraunhofer transform simulation of an SLM was built (Appendix B: SLMSimulation2D.py) to describe the 0.8π limited Zernike polynomial phase masks interacting with natural Zernike aberrations. The natural aberrations go through a full 2π phase, as we have seen in the complete Fraunhofer transform of the system in equation (18). In the simulation, we introduced the aberration $c_7 Z_3^1$, either through the SLM or a natural aberration, with $b7 = \frac{100.0}{255}$ for the SLM coefficient or $a7 = \frac{100.0}{255}$ for the natural aberration. The results can be seen in Figure 10. In the case that only the SLM Zernike polynomials are used, with the 0.8π limit phase shift, the simulation demonstrates that each diffraction order has a different point spread function. But in the case that only the natural Zernike aberrations are used, we see that each diffraction order appears the same. Going back to the experimental apparatus, we then used the series iteration method on the +2, +1,



(a) **(b)**

FIG. 10: Flipped PSF of Diffraction Orders. A simulation of the SLM or natural aberrations was preset with $b7 = \frac{100.0}{200}$ (a) or $a7 = \frac{100.0}{200}$ (b). The Zernike polynomial is Z_3^1, which corresponds to horizontal coma. If you look closely, the PSF of the +1 and -1 diffraction orders appear flipped and different for the 0.8π phase limited SLM aberration (a), while the natural aberration PSF was the same for each diffraction order in (b).

-1, and -2 diffraction orders while there was no soda bottle in the beam path, which was essentially another control run of the method. We compiled the coefficients found that had minimized the spot metric for each diffraction order. In Figure 11, we found that the Zernike coefficients applied to the SLM had minimized

the spot metric at values which were opposite of each other when observed on the opposite diffraction order (i.e +1 and -1, or +2 and -2), but different in magnitude when compared to the same side (i.e +2 and +1, or -2 and -1).

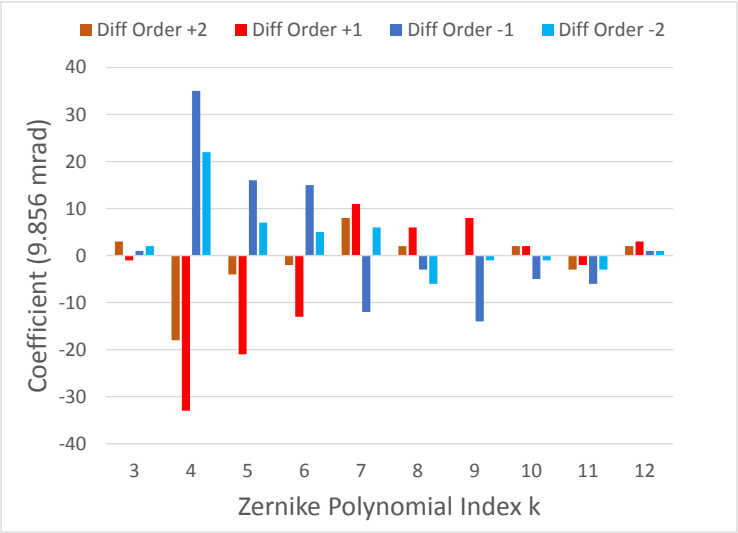


FIG. 11: Flipped Coefficients of Diffraction Orders. Starting at $k = 3$, the series iteration method found that most of the significant coefficients had flipped values on the opposing diffraction order. For example, $c_4 = -33, -35$ for +1 and -1, while $c_4 = -18, -22$ for +2 and -2.

In effect, they disagreed on what the actual aberration was that they were minimizing in equation (18), even though the spot metric was minimized for their respective diffraction order.

Going back to the simulation, we used the same coefficient values for the SLM and opposite coefficient values for the natural aberrations; we make the assumption that the SLM Zernike phase mask is canceling with the aberrations. In Figure 12, the results showed a minimization of the spot metric for the +1 diffraction order if the SLM is turned on, but no minimization in other diffraction orders. For Figure 12, the final S_{end} in all cases was around $S_{end} \approx 10.5$, with the highest $S_{start} = 25.6$ from the set of +1 diffraction order coefficients in Figure 11. However, the +1 diffraction order was minimized for all sets of coefficients from Figure 11, under the assumption that the natural Zernike aberrations had coefficients that were the opposite of the SLM Zernike polynomials. The other diffraction orders always appear smeared, while the +1 diffraction order maintains the PSF of a rectangular aperture free of aberrations (a cross), which is what we would expect to observe from the rectangular aperture on the SLM if the laser beam intensity was high enough.

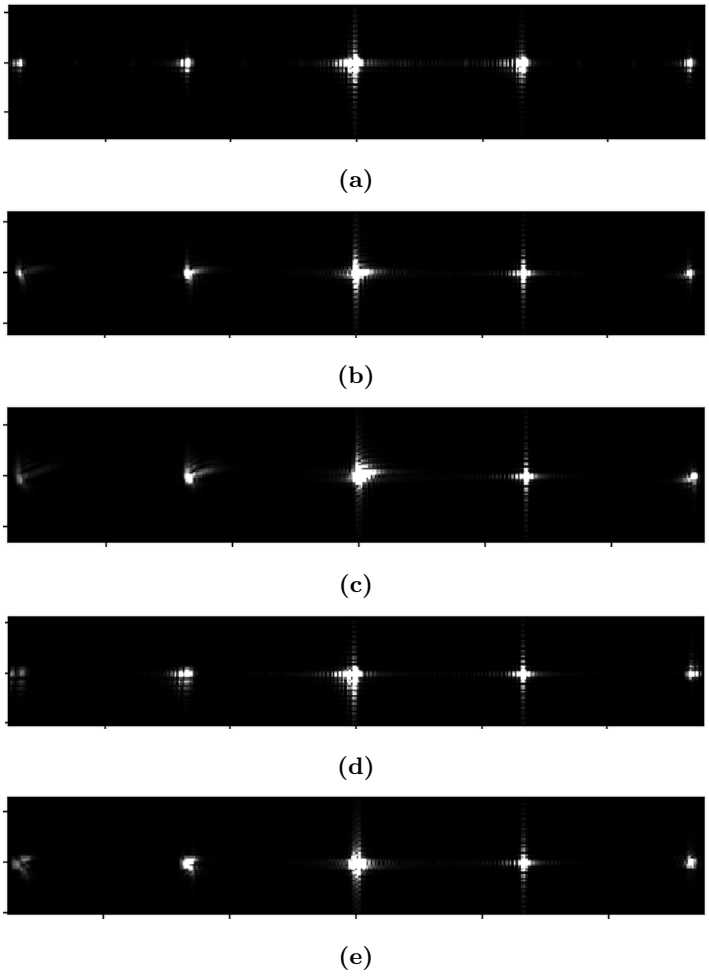


FIG. 12: Simulation of SLM Phase Canceling With Natural Aberrations. Counting from left to right, the +1 diffraction order is the 4th bright spot of light. Simulation of the SLM turned off with natural Zernike aberrations set to the -1 diffraction order coefficients found in Figure 11 (a). SLM now turned on, with opposite coefficient values for SLM and natural aberrations found from +2 diffraction order in Figure 11, $\%\Delta S_{+2} = 34\%$ (b). With aberrations found from +1 diffraction order in Figure 11, $\%\Delta S_{+1} = 59\%$ (c). With aberrations found from -1 diffraction order in Figure 11, $\%\Delta S_{-1} = 44\%$ (d). With aberrations found from -2 diffraction order in Figure 11, $\%\Delta S_{-2} = 45\%$ (e).

The simulation results seem to show that the +1 diffraction order is the correct order to determine the opposite coefficients of the natural Zernike aberrations, and the experimental data shows that the +1 diffraction order measured the most distortions, as it had the highest $\%\Delta S$ of about 59%. However, another SLM capable of 2π phase depth would be required to experimentally verify the result. Otherwise, the simulation and experimental results seem to agree that the coefficients are flipped between opposing diffraction orders and that each diffraction order has its own set of coefficients which minimize the associated spot metric.

The simulation and experimental results agree, that diffraction orders can become individually less aberrated, but not globally less aberrated for the other orders in the case of a 0.8π phase limited SLM. But more importantly, the simulation seems to demonstrate that you could potentially **model** the aberrations with a 0.8π phase limited SLM, but only if you minimize the spot metric of the +1 diffraction order. One final, but very interesting observation, is that the $\%\Delta S$ found in the simulations from Figure 12 all have very similar strengths to the experimental $\%\Delta S$ of the bottle between L1 and the SLM, as well as the control in Figure 8 (a).

V. CONCLUSION

The original motivation was that different scientific disciplines encounter aberrations on the wavefronts of light, and that there would be a benefit to removing or characterizing these aberrations. Many tools and control methods have been adopted to address aberrated wavefronts, for example, the SLM we have used in this research project or deformable mirrors used in other projects. We attempted to use a LCOS SLM, limited to a phase shift depth of only 0.8π, along with a blazed phase profile and Zernike polynomial phase mask, to remove aberrations from a $630nm$ laser beam. Using a plastic bottle as an aberration source, we were able to determine the effectiveness of the system using a spot metric of different diffraction orders.

We conclude that the system is capable of consistently removing aberrations with 13 different Zernike polynomial phase profiles, but only from a particular diffraction order. However, the results did not correspond to other diffraction orders simultaneously, but could be reproduced for each one separately. That meant the global image of the system was not fully removed of aberrations. But one workaround proposed is that we could pass an image to that diffraction order if its PSF was narrow enough to not overlap with other diffraction orders, which would place limitations on the kinds of images transported by the laser beam. To recover the image from the diffraction order, one would need a 4F optical apparatus in a future experiment, with a filter placed on the other diffraction orders.

We also ran a simulation of the optical apparatus and found that our experimental results agreed with the simulation results, such that 0.8π SLM is capable of removing aberrations in only one diffraction order at a time. Our results also conclude that the SLM just needs to flip the phase of the Zernike polynomials that remove aberrations from one diffraction order, to maximize the opposing diffraction order (i.e from +1 to -1).

The other part of the motivation addressed was whether or not we could **model** Zernike polynomial aberrations with a 0.8π limited SLM, rather than just **remove** them. The experimental results were inconclusive, but the simulation revealed the potential ability to **model** the actual Zernike aberrations on the wavefront at the point of reflection off the SLM. According to the simulation, the +1 diffraction order (which is the largest using our choice of blaze) becomes unaberrated when the SLM's Zernike polynomial phase mask has the exact opposite polynomial of the true natural aberration. If the experimental system perfectly matched the simulation, then the Zernike aberrations removed from the +1 diffraction order are the actual natural aberrations.

Some future work should be done in isolating a diffraction order with aberrations removed and passing it through another lens (like a 4F setup). We could analyze the image quality and what limitations exist on the types of images transmitted. Using the same Zernike polynomial phase masks, future researchers should also consider testing a 0.8π LCOS SLM with a full 2π phase depth capable SLM. The researchers should

copy the aberrations detected on the +1 diffraction order using the 0.8π SLM to the phase profile of the 2π phase SLM. If more diffraction orders are simultaneously improved with the phase profile, then we could conclusively state that a 0.8π phase SLM is capable of modeling Zernike aberrations.

VI. ACKNOWLEDGEMENTS

Thank you Professor David H. McIntyre for advising me on how I should approach my research, and I enjoyed discussing other interesting concepts with you, which kept me motivated. A special thank you to Professor Janet Tate for giving me an early start on my thesis, despite my work not having a well defined sense of direction yet. I also want to thank Ethan Minot, as he was my thesis professor for 2 terms, which helped me understand the rules and guidelines for formatting, something which I am sure all of us dread. Finally, thank you to the Physics Department of Oregon State University, for giving me the opportunity to play with tools I thought I would never get a chance to use.

[1] J. M. Beckers, "Adaptive optics for astronomy: principles, performance, and applications," *Annual review of astronomy and astrophysics* **31**(1), 13-62 (1993).

[2] I. Kozak, "Retinal imaging using adaptive optics technology," *Saudi journal of ophthalmology : official journal of the Saudi Ophthalmological Society* **28**(2), 117-122 (2014).

[3] J. C. Wyant and K. Creath, in *Applied Optics and Optical Engineering,* edited by J. C. Wyant and R. R. Shannon (Elsevier Science & Technology Books, 1992), Vol. 11, pp. 1-53.

[4] A. Tahmasbi, F. Saki and S. B. Shokouhi, "Classification of benign and malignant masses based on zernike moments," *Computers in Biology and Medicine* **41**(1), 726-735 (2011).

[5] C. J. R. Sheppard, in *Handbook of Biomedical Optics,* edited by D. A. Boas, C. Pitris, N. Ramanujam (CRC Press, 2011), pp. 11-31.

[6] E. Hecht, *Optics* (Pearson Education Inc., 2017), 5th ed.

[7] K. T. Mcdonald, "Axicon Gaussian Laser Beams," *ArXiv Physics e-prints.* https://arxiv.org/abs/physics/0003056 (2000).

[8] M. A. Cibula and D. H. McIntyre, "General algorithm to optimize the diffraction efficiency of a phase-type spatial light modulator," *Opt. Lett.* **38**(15), 2767-2769 (2013).

Appendix A: SLMSimulator1D.py

SLMSimulator1D.py is a python script capable of simulating a 1 dimensional (x-axis) Fraunhofer diffraction pattern if we map a blazed grating and Zernike polynomials to the SLM. In addition, it simulates natural Zernike aberrations and their interaction with the SLM.

BEGIN CODE:

```
#AUTHOR: Aaron Goschie
#Physics Department, Oregon State University

import numpy as np
import matplotlib.pyplot as pl
import csv

#config
output_graph_excel = False
compute_spot_metric = True
display_cycle = False

resolution_depth = 10 #bigger = less resolution
depth_factor=0.8 #x pi phase

spot_metric_start = 1.6 * resolution_depth
spot_metric_end = 1.8 * resolution_depth

#aberration Zernike coefficients (always through 2pi phase)
a3=0
a4=0
a5=0
a6=-0.3
a7=0
a8=0
a9=0
a10=0
```

```
a11=0
a12=0

#slm Zernike coefficients (defaults to zero for "compute spot metric")
b3=0
b4=0
b5=0
b6=0.3
b7=0
b8=0
b9=0
b10=0
b11=0
b12=0

#globals

#Discretize time t
t0 = -30000 #t0=-30000 #t0=-1000.
dt = 0.006 * resolution_depth #dt=0.006 #dt=0.0001

#dt is equivalent to 9 micrometers
tfactor = dt / 9.0

#physical
lamb=0.63 #micrometer wavelength
lens=200000 #micrometer focal length
focal_factor=(lens*tfactor*tfactor*lamb / 2.0) / np.pi
#f/k due to fraunhofer form, this way fourier transform is equivalent

#physics functions
def blazedGrating(t):
    return np.mod(t / (50 * tfactor), 1)
```

```
def ZP_0_2(t, r):
    return(-1 + 2*(t/r)**2)
def ZP_2_2(t, r):
    return((t/r)**2)
def ZP_N2_2(t, r):
    return(0)
def ZP_1_3(t, r):
    return ((t/r)*(-2.0 + 3.0*((t/r)**2)))
def ZP_N1_3(t, r):
    return(0)
def ZP_0_4(t, r):
    return(1-6*((t/r)**2)+6*((t/r)**4))
def ZP_3_3(t, r):
    return((t/r)**3)
def ZP_N3_3(t, r):
    return(0)
def ZP_2_4(t, r):
    return(-3*((t/r)**2) + 4*((t/r)**4))
def ZP_N2_4(t, r):
    return(0)


def zernikeCombination(t, r, c3, c4, c5, c6,
                       c7, c8, c9, c10, c11, c12):
    return np.mod((np.heaviside(t+r, 1) - np.heaviside(t - r, 1))*
                  (c3*ZP_0_2(t,r) +
                   c4*ZP_2_2(t,r) +
                   c6*ZP_1_3(t, r) +
                   c8*ZP_0_4(t,r) +
                   c9*ZP_3_3(t,r) +
                   c11*ZP_2_4(t,r)), 1)

def aberration(t, r, c3, c4, c5, c6,
               c7, c8, c9, c10, c11, c12):
```

```
    return 2j*np.pi*zernikeCombination(t, r, c3, c4, c5, c6,
    c7, c8, c9, c10, c11, c12)

def phaseMask(t, r, c3, c4, c5, c6, c7, c8, c9, c10, c11, c12):
    return np.mod(blazedGrating(t) + zernikeCombination(t, r, c3, c4, c5, c6,
        c7, c8, c9, c10, c11, c12), 1)

def windowFunc(t, r):
    return (np.heaviside(t + r, 1) - np.heaviside(t - r, 1))

if not compute_spot_metric :
    #create time segments
    t=np.arange(t0,-t0,dt)
    lensTransformPhaseFactor = -1j*np.exp(2j*2*np.pi*lens
        / lamb)/(lamb * tfactor * lens * tfactor)

    #Aperture function of SLM
    f=windowFunc(t, (dt*1024.0 / 2.0))*np.exp(
        1j*depth_factor*np.pi*phaseMask(t, dt*768.0/2.0, b3, b4, b5, b6,
            b7, b8, b9, b10, b11, b12)
        + aberration(t, dt*768.0 / 2.0, a3, a4, a5, a6,
            a7, a8, a9, a10, a11, a12))

    #FFT
    F=np.sqrt(2*np.pi)*lensTransformPhaseFactor*np.fft.fft(f)


#
#To make it equivalent to continuous fourier transform,
#we use the phase factor methods described on www.stackoverflow.com
#URL: https://stackoverflow.com/questions/
#24077913/discretized-continuous-fourier-transform-with-numpy
#AUTHOR: thomasfermi
#AUTHOR URL: https://stackoverflow.com/users/2609987/thomasfermi
```

```
#frequency normalization factor is 2*np.pi/dt
    w = np.fft.fftfreq(f.size)*focal_factor*2*np.pi/dt
#dt*np.exp(-1j*w*t0/focal_factor)/(np.sqrt(2*np.pi))
    F*=dt/(np.sqrt(2*np.pi))
#get a discretisation of the continuous Fourier transform
#by multiplying F by a phase factor
#



    print(w.size)
    print(F.size)
    print(w[int(f.size/2)])

    #show focal plane
    pl.scatter(w,np.abs(F)**2, s=0.5,color="r")

    pl.gca().set_xlim(-5*resolution_depth,5*resolution_depth)
    pl.gca().set_ylim(0, 2200/(resolution_depth**2))
    pl.xlabel('Relative Spacing', fontsize=12)
    pl.ylabel('Intensity (W/m^2)', fontsize=12)
    pl.show()
    pl.close()

else :
    frequencies = []
    graphs = []
    spot_metrics = []

    aperture_window = []
    aperture_natural = []
    phase_conversion = []

    #create time segments
    t=np.arange(t0,-t0,dt)
```

```
lensTransformPhaseFactor = -1j*np.exp(2j*2*np.pi*lens
    / lamb)/(lamb * tfactor * lens * tfactor)

#frequency normalization factor is 2*np.pi/dt
w = np.fft.fftfreq(t.size)*focal_factor*2*np.pi/dt
begin = (int(int(w.size/2)*spot_metric_start/w[int(w.size/2)-1]))
end = (int(int(w.size/2)*spot_metric_end/w[int(w.size/2)-1]))
print(begin)
print(end)
phase_conversion=dt/(np.sqrt(2*np.pi))

#get a discretisation of the continuous Fourier transform
#by multiplying F by a phase factor
aperture_window = windowFunc(t, (dt*1024.0 / 2.0))
aperture_natural = aperture_window*np.exp(
    aberration(t, dt*768.0 / 2.0, a3, a4, a5, a6,
    a7, a8, a9, a10, a11, a12))

print("loading...")
for i in range(100):
        #Aperture function of SLM
        print(i)
        f=aperture_natural*np.exp(
            1j*depth_factor*np.pi*phaseMask(t, dt*768.0/2.0,
            b3, b4, b5, (i-50)/100,
            b7, b8, b9, b10, b11, b12))
        F=np.sqrt(2*np.pi)*lensTransformPhaseFactor*np.fft.fft(
        f)*phase_conversion #FFT
        graphs.append(np.abs(F)**2)
        #spot metric calculation, storage
        spot_metrics.append(np.power(np.sum(graphs[i][begin:end]), 2)
                            / np.sum(np.power(graphs[i][begin:end], 2)))

if display_cycle :
```

```
    for i in range(100):
            pl.gca().set_xlim(-5*resolution_depth,5*resolution_depth)
            pl.gca().set_ylim(0,2200/(resolution_depth**2))
            pl.scatter(w,graphs[i], s=0.5, color="r",
                label="Zernike 0,2, Coeff = " + str(((i-50)/100.0)))
            pl.xlabel("x-axis")
            pl.ylabel("Intensity")
            pl.legend()
            pl.draw()
            pl.pause(0.1)
            pl.clf()

pl.gca().set_xlim(-0.5,0.5)
pl.plot(np.arange(-0.5, 0.5, 0.01), spot_metrics, label="Spot Metric")
pl.xlabel("Coeff N")
pl.ylabel("Spot Metric")
pl.legend()
pl.show()
pl.close()
```

Appendix B: SLMSimulator2D.py

SLMSimulator2D.py is a python script capable of simulating the full Fraunhofer diffraction pattern in a CCD camera if we map a blazed grating and Zernike polynomials to the SLM. In addition, it simulates natural Zernike aberrations and their interaction with the SLM.

BEGIN CODE:

```
#AUTHOR: Aaron Goschie
#Physics Department, Oregon State University

import numpy as np
import matplotlib.pyplot as pl
import matplotlib.cm as cm
import csv

#config
output_graph_excel = False
compute_spot_metric = False
display_cycle = False

resolution_depth = 1/0.006#4000 #bigger = less resolution
depth_factor=0.8 #x pi phase
power_factor=0.008
CCDmin=0
CCDmax=0.04

spot_metric_startX = 55
spot_metric_endX = 75
spot_metric_startY = -10
spot_metric_endY = 10

#grating spacing (micrometers)
grating=210
grating_max=1 #0 to 1
```

```
#aberration Zernike coefficients (always through 2pi phase shift)

a3=0
a4=0
a5=0
a6=0
a7=0
a8=0
a9=0
a10=0
a11=0
a12=0

#slm Zernike coefficients (defaults to zero for "compute spot metric")

b3=0
b4=0
b5=0
b6=0
b7=0
b8=0
b9=0
b10=0
b11=0
b12=0

#globals
#gaussian width factor
gaussian_width_factor = 90000000
#Discretize time t
t0 = -1000#-2250#-30000 #t0=-30000 #t0=-1000.
dt = 0.006 * resolution_depth #dt=0.006 #dt=0.0001
```

```
#dt is equivalent to 9 micrometers
tfactor = dt / 9.0

#physical
lamb=0.63 #micrometer wavelength
lens=200000 #micrometer focal length
focal_factor=(lens*tfactor*tfactor*lamb / 2.0) / np.pi
#f/k due to fraunhofer form, this way fourier transform is equivalent

#physics functions
zernike_window=[[]]
blaze_phase=[[]]
def blazedGrating(t1, t2):
    return (np.mod(t1 / (grating * tfactor), grating_max)*np.exp(0*t2))

def ZP_0_2(t1,t2, r):
    return(-1 + 2*((t1/r)**2 + (t2/r)**2))
def ZP_2_2(t1,t2, r):
    return(((t1/r)**2) - ((t2/r)**2))
def ZP_N2_2(t1,t2, r):
    return(2*(t1/r)*(t2/r))
def ZP_1_3(t1,t2, r):
    return ((t1/r)*(-2.0 + 3.0*(((t1/r)**2) + ((t2/r)**2))))
def ZP_N1_3(t1,t2, r):
    return ((t2/r)*(-2.0 + 3.0*(((t1/r)**2) + ((t2/r)**2))))
def ZP_0_4(t1,t2, r):
    return(1-6*(((t1/r)**2) + ((t2/r)**2))+6*(((t1/r)+(t2/r))**2))
def ZP_3_3(t1,t2, r):
    return(((t1/r)**3)-3*(t1/r)*(t2/r)**2)
def ZP_N3_3(t1,t2, r):
    return(3*((t1/r)**2)*(t2/r) - ((t2/r)**3))
def ZP_2_4(t1,t2, r):
    return(-3*((t1/r)**2) + 4*((t1/r)**4) + 3*((t2/r)**2) - 4*((t2/r)**4))
def ZP_N2_4(t1,t2, r):
```

```
    return(2*(t1/r)*(t2/r)*(-3 + 4*(((t1/r)**2)+((t2/r)**2))))


def zernikeCombination(t1, t2, r, c3, c4, c5, c6,
                       c7, c8, c9, c10, c11, c12):
    #return np.mod(np.heaviside(-(t1**2) - (t2**2) + (r**2), 1) * (
    return np.mod((
                    c3*ZP_0_2(t1,t2,r) +
                   c4*ZP_2_2(t1,t2,r) +
                   c5*ZP_N2_2(t1,t2,r) +
                   c7*ZP_1_3(t1,t2, r) +
                   c6*ZP_N1_3(t1,t2,r) +
                   c8*ZP_0_4(t1,t2,r) +
                   c10*ZP_3_3(t1,t2,r) +
                   c9*ZP_N3_3(t1,t2,r) +
                   c11*ZP_2_4(t1,t2,r) +
                   c12*ZP_N2_4(t1,t2, r)), 1)

def aberration(t1, t2, r, c3, c4, c5, c6,
               c7, c8, c9, c10, c11, c12):
    return 2j*np.pi*zernikeCombination(t1, t2, r, c3, c4, c5, c6,
                                       c7, c8, c9, c10, c11, c12)

def phaseMask(t1, t2, r, c3, c4, c5, c6,
              c7, c8, c9, c10, c11, c12):
    return np.mod(blazedGrating(t1, t2) +
        zernikeCombination(t1, t2, r, c3, c4, c5, c6,
                           c7, c8, c9, c10, c11, c12), 1)

def windowFunc(t1, t2, r1, r2):
    return ((np.heaviside(t1 + r1, 1) - np.heaviside(t1 - r1, 1)) *
            (np.heaviside(t2 + r2, 1) - np.heaviside(t2 - r2, 1)))

if not compute_spot_metric :
```

```
    #create time segments
    t=np.arange(t0,-t0,dt)
    y=t[ : ,None]
    x=t[None, : ]

    zernike_window = np.heaviside(-(x**2) - (y**2) + ((dt*768.0/2.0)**2), 1)
    blaze_phase = blazedGrating(x, y)

    lensTransformPhaseFactor = -1j*np.exp(2j*2*np.pi*lens
        / lamb)/(lamb * tfactor * lens * tfactor)

    gaussianProfile=np.exp(
        -1*(x**2 + y**2)/(dt*gaussian_width_factor)
        )
    aperture_window=power_factor*windowFunc(x, y, (dt*1024.0 / 2.0),
    (dt*768.0 / 2.0))*gaussianProfile

    aperture_natural=aperture_window*np.exp(
        aberration(x, y, dt*768.0 / 2.0, a3, a4, a5, a6,
            a7, a8, a9, a10, a11, a12))

    #Aperture function of SLM
    print("creating aperture...")
    f=aperture_natural*np.exp(
        1j*depth_factor*np.pi*phaseMask(x, y, dt*768.0/2.0, b3, b4, b5, b6,
            b7, b8, b9, b10, b11, b12))

    #FFT
    print("creating transform...")
    F=np.sqrt(2*np.pi)*lensTransformPhaseFactor*np.fft.fft2(f)


#
#To make it equivalent to continuous fourier transform,
```

```
#we use the phase factor methods described on www.stackoverflow.com
#URL: https://stackoverflow.com/questions
#/24077913/discretized-continuous-fourier-transform-with-numpy
#AUTHOR: thomasfermi
#AUTHOR URL: https://stackoverflow.com/users/2609987/thomasfermi
#frequency normalization factor is 2*np.pi/dt
    w = np.fft.fftfreq(f[0].size)*focal_factor*2*np.pi/dt
#get a discretisation of the continuous Fourier transform
#by multiplying F by a phase factor
    F*=(dt**2)/(np.sqrt(2*np.pi))
#
#

    print("displaying...")
    pl.imshow(phaseMask(x, y, dt*768.0/2.0, b3, b4, b5, b6,
            b7, b8, b9, b10, b11, b12), extent=(-512,512,-384,384),
    interpolation=None, cmap=cm.gray, vmin=0, vmax=1)
    pl.show()

    F = np.fft.fftshift(F)

    beginX = int((spot_metric_startX / (w[1]-w[0])) + (
    gaussianProfile[0].size / 2))
    endX = int((spot_metric_endX / (w[1]-w[0])) + (
    gaussianProfile[0].size / 2))
    beginY = int((spot_metric_startY / (w[1]-w[0])) + (
    gaussianProfile[0].size / 2))
    endY = int((spot_metric_endY / (w[1]-w[0])) + (
    gaussianProfile[0].size / 2))
    print(np.power(np.sum((0.5*np.abs(F)**2)[beginY:endY, beginX:endX]), 2)
            / np.sum(((0.5*np.abs(F)**2)[beginY:endY, beginX:endX])**2))

    pl.clf()
    pl.imshow(0.5*np.abs(F)**2, extent=(w.min(),w.max(),w.min(),w.max()),
```

```
    interpolation=None, cmap=cm.gray, vmin=CCDmin, vmax=CCDmax)
    pl.xlim(-5*resolution_depth/6,5*resolution_depth/6)
    pl.ylim(-5*resolution_depth/6,5*resolution_depth/6)
    pl.show()

else :
    #spot metric stuff goes here
    frequencies = []
    graphs = []
    spot_metrics = []
    spot_metric_min=[10000000,0]

    aperture_window = []
    aperture_natural = []
    phase_conversion = []

    #create time segments
    t=np.arange(t0,-t0,dt)
    y=t[ : ,None]
    x=t[None, : ]

    zernike_window = np.heaviside(-(x**2) - (y**2) + (dt*768.0/2.0**2), 1)
    blaze_phase = blazedGrating(x, y)

    lensTransformPhaseFactor = -1j*np.exp(2j*2*np.pi*lens
        / lamb)/(lamb * tfactor * lens * tfactor)

    gaussianProfile=np.exp(
        -1*(x**2 + y**2)/(dt*gaussian_width_factor)
        )
    aperture_window=power_factor*windowFunc(x, y, (dt*1024.0 / 2.0),
    (dt*768.0 / 2.0))*gaussianProfile

    aperture_natural=aperture_window*np.exp(
```

```
    aberration(x, y, dt*768.0 / 2.0, a3, a4, a5, a6,
        a7, a8, a9, a10, a11, a12))

w = np.fft.fftfreq(gaussianProfile[0].size)*focal_factor*2*np.pi/dt
sm_shift=0.5
if (gaussianProfile[0].size % 2) != 0 :
    sm_shift = 0
beginX = int((spot_metric_startX / (w[1]-w[0])) + (
gaussianProfile[0].size / 2))
endX = int((spot_metric_endX / (w[1]-w[0])) + (
gaussianProfile[0].size / 2))
beginY = int((spot_metric_startY / (w[1]-w[0])) + (
gaussianProfile[0].size / 2))
endY = int((spot_metric_endY / (w[1]-w[0])) + (
gaussianProfile[0].size / 2))
print(beginX)
print(endX)
print(beginY)
print(endY)
phase_conversion=(dt**2)/(np.sqrt(2*np.pi))

print("loading...")
for i in range(100):
        #Aperture function of SLM
        print(i)
        f=aperture_natural*np.exp(
            1j*depth_factor*np.pi*phaseMask(
            x, y, dt*768.0/2.0, b3, (i)/255, b5, b6,
            b7, b8, b9, b10, b11, b12))
        F=np.sqrt(
        2*np.pi)*lensTransformPhaseFactor*np.fft.fft2(
        f)*phase_conversion #FFT
        F = np.fft.fftshift(F)
        graphs.append((0.5*(np.abs(F)**2)))
```

```
            #spot metric calculation, storage
            spot_metrics.append(
                np.power(np.sum(graphs[i][beginY:endY, beginX:endX]), 2)
                    / np.sum((graphs[i][beginY:endY, beginX:endX])**2))
            if spot_metric_min[0] > spot_metrics[i] :
                spot_metric_min[0] = spot_metrics[i]
                spot_metric_min[1] = i


    print(spot_metric_min[1])
    pl.gca().set_xlim(-0/255,50/255)
    pl.gca().set_ylim(0,60)
    pl.plot(np.arange(-0/255, 100/255, 1/255),
    spot_metrics, label="Spot Metric")
    pl.xlabel("Coeff N")
    pl.ylabel("Spot Metric")
    pl.legend()
    pl.show()
    pl.close()
```

END CODE