Examination of a lensless setup for reflective inverse diffusion of light

THESIS

John Nguyen, 1st Lt, USAF

AFIT/ENP/21-M-128

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/ENP/21-M-128

EXAMINATION OF A LENSLESS SETUP FOR REFLECTIVE INVERSE DIFFUSION OF LIGHT

THESIS

Presented to the Faculty

Department of Engineering Physics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Applied Physics

John Nguyen, B.S.

1st Lt, USAF

26 March 2021

AFIT/ENP/21-M-128

EXAMINATION OF A LENSLESS SETUP FOR REFLECTIVE INVERSE DIFFUSION OF LIGHT

THESIS

John Nguyen, B.S.
1st Lt, USAF

Committee Membership:

Lt Col Kenneth W. Burgi
Chairman

Michael A. Marciniak, PhD
Member

Anthony Franz, PhD
Member

AFIT/ENP/21-M-128

Abstract

Reflective inverse diffusion uses spatial light modulators to shape an incident wavefront so that when the wavefront interacts with some diffuse scattering sample, the reflected light will constructively interfere at a single focus. By combining reflective inverse diffusion with beam steering, it is possible to image objects around corners and obstacles. Traditionally, reflective inverse diffusion uses a focal plane system where a lens would focus light modified by the spatial light modulator onto the scattering sample. The focal plane system would also have to compensate for the Fourier transform when beamsteering with the lens. Removal of this lens from the system allows any modulation of the incident light to be directly projected onto a scattering sample. Additionally, limitations in the system from the 2D Fourier transform of the lens would be lifted in a lensless system and allow for more light to be focused at a single spot. This thesis examines the efficacy of the lensless system against the focal plane system in achieving reflective inverse diffusion and beamsteering. Although the lensless setup outperformed its counterpart by being able to focus more energy in a region, the lensless setup failed to achieve the same beam steering capabilities as the focal plane system. Understanding the flaws behind the lensless system will be instrumental in creating a setup that will achieve reflective inverse diffusion while possessing the capability to effectively beamsteer.

Table of Contents

Page

List of Figures

List of Abbreviations

Abbreviation		Page
AFIT	Air Force Institute of Technology	2
SLM	Spatial Light Modulator	9
LCoS	Liquid Crystal on Silicon	9
CCD	Charged Coupled Device	13
PCIe	Peripheral Component Interconnect Express	16
LUT	Look Up Table	16

EXAMINATION OF A LENSLESS SETUP FOR REFLECTIVE INVERSE DIFFUSION OF LIGHT

I. Introduction

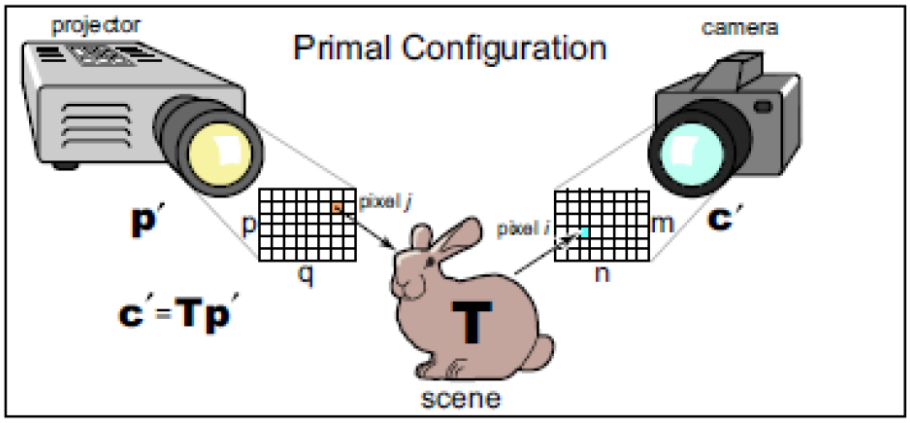
1.1 Motivation

Conventional imaging requires a direct line of sight in which light travels from the surface of an object to the imaging system. If there are obstacles that obstruct the direct line of sight then this problem can be remedied through the use of a mirror. Mirrors will specularly reflect light to form clear images forgoing the need for a direct line of sight. It is also important to note that these mirrors must also be comprised of non-absorptive media otherwise any incident light would be absorbed rather than reflected. Objects whose surfaces are diffuse will not form clear images due to the fact that upon reflection, the light is scattered in all directions. To alleviate these hindrances, the principle of wavefront shaping can be applied to image objects without direct line of sight by using reflections off surfaces that are diffusely scattering. This principle works by altering an incident wavefront so that when the altered wavefront interacts with the scattering surface, light will constructively interfere and focus rather than scatter. Image formation would no longer need a direct line of sight making it possible to see around corners. This technology would not only provide a great tactical advantage in the hands of the warfighter but would also have wider applications.

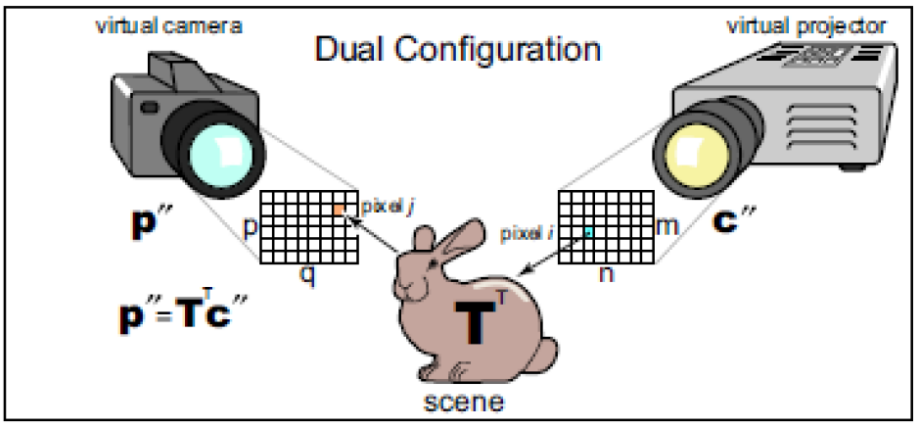
1.2 Dual and Indirect Photography

Dual photography is a photographic technique which interchanges the light source of a scene and the camera [1]. By interchanging the light source and the camera, it is possible to capture details hidden to the camera but available to the point of view of the light source illuminating the scene. This technique can be seen as a direct consequence of the principle of Helmholtz reciprocity which is the idea that light traveling from a source to an observer can be reversed without any changes in its transport properties [1]. In essence, dual photography uses a pixelated light source such as a projector to illuminate a scene with many pixels one at a time while simultaneously using a camera to capture how each particular pixel interacts with the system. After capturing an image for each light source pixel, an algorithm can be applied to generate an image as though it was captured from the perspective of the pixelated light source and illuminated from the position of the camera. Figure 1 demonstrates how a scene can be captured from different angles using this technique. As a result of Helmholtz reciprocity, the transport matrices of each configuration are the transpose of another one which is how the algorithm is able to recreate the scene in the interchanged configuration in post processing.

The requirement of both a camera and a pixelated light source prevents this photographic technique from having any practical applications. If a scene needed to be observed at a particular angle, it would be easier to simply use an additional camera in place of a light source at that location. To further expand on the practical applications of dual photography, indirect photography was studied at the Air Force Institute of Technology (AFIT) [2]. Indirect photography is a method that relieves the burden of direct line of sight for the light source by co-locating the projector and the camera and using radiometric models of diffuse reflections to capture details hidden



(a) 1a. Primal Configuration



(b) 1b. Dual Configuration

Figure 1. A camera and a projector can be interchanged in post processing to reveal details hidden to the camera originally. 1a is the original or primal configuration with its set of coordinates and transport matrix which represents the scene, T. 1b is the interchanged or dual configuration with its own unique set of coordinates and its transport matrix which is the transpose of T, reprinted with permission [2].

to the point of view of both camera and projector [2, 3]. In theory when indirect photography is combined with wavefront shaping, scenarios can become reduced to a dual photography problem and imaging becomes possible without the need for a direct line of sight.

1.3 Wavefront Shaping

Wavefront shaping is the method of constructing a wavefront that is perfectly matched to a particular surface such that the multiple scattering of light can be focused after diffuse reflection [4]. Although the scattering of light is commonly thought of as a stochastic process akin to a "random walk", scattering is actually a linear and deterministic process. It is thus possible to analyze a surface's effect on an incident wavefront and tailor the wavefront so that light will focus following its interaction with the surface. The focusing of light following this diffuse scattering is termed inverse diffusion and can be further extended to other light transport processes such as transmissive inverse diffusion and reflective inverse diffusion [2]. By shaping the incident wavefront such that inverse diffusion is achieved and light is focused after scattering off some surface, the scattering surface is now effectively a projector that has line of sight at a particular angle of interest without the need of equipment having to be physically present at that location. Wavefront shaping removes the limitations of dual photography and allows it to have profound impact when applied.

1.4 Objective

The concept of dual photography involves illumination by a single pixel of light via a projector. When combined with inverse diffusion and indirect photography, a scattering reflective surface is effectively turned into that projector. However, to simulate the projector illuminating pixel by pixel, it is necessary to either create an entirely new and different wavefront shape that will focus the scattering light at a different location or beam steer the original optimized spot to another location. The objective of this research is to validate whether a lensless setup can produce more intense focused spots that have the capability of being beam steered than their counterpart setups.

II. Background

2.1 Inverse Diffusion

The propagation of light through some turbid medium scatters in multiple directions. If the light is coherent, then a "speckle" interference pattern will form after scattering. These speckle patterns are the result of the interference between the incident optical field and the scattering medium which can be represented as some complex field of Fourier components [5]. The representation of the medium as a complex field leads to the idea of wavefront shaping the incident light to effectively turn the scattering medium into either a lens or mirror depending on the usage needed [5, 4, 6, 7].

2.2 Image Plane vs. Focal Plane Setup vs. Lensless Setups

Analysis taken from previous studies demonstrated the effectiveness of using a focal plane setup and an image plane setup in the reflective inverse diffusion of light [2, 8]. An example of these particular types of setups can be seen in Figures 2 and 3. Focal plane systems place the scattering sample at the focal point of the lens while image plane systems place the sample away from the focal point to project the demagnified image onto the scattering sample. The focal plane system was shown to create a wavefront mask that would achieve reflective inverse diffusion and focus light after scattering [8]. However, using a lens that focuses the modulated light onto the sample imposes some practical limitations on the intensity of the optimized spot and limitations on beam steering. The segments of the SLM act as apertures that produce fringes due to the 2D Fourier transform of the lens [2]. SLM segments that are the furthest apart produce the narrowest fringes, while segments that are right next to each other will produce the widest set of fringes. It is these set of narrow

fringes that define the upper limit of spatial frequencies and how precisely the SLM can conjugate the scattering. The limited amount of space on the SLM results in a high ratio of narrow fringes to wide fringes.

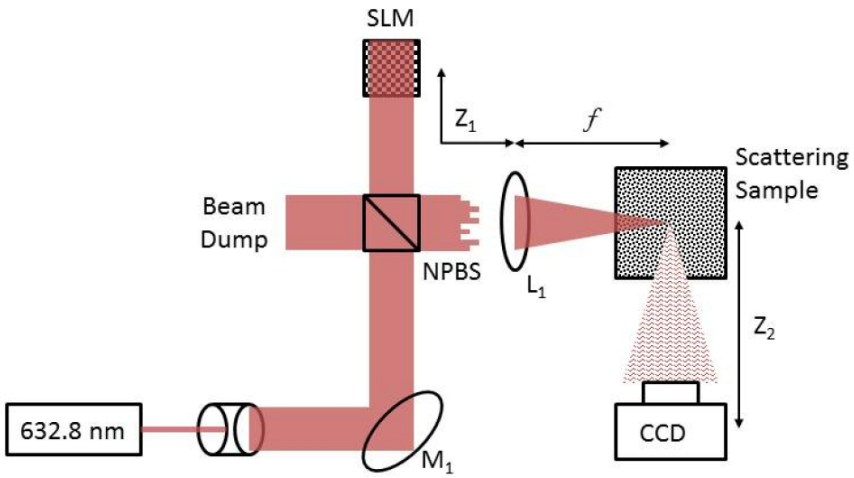


Figure 2. A demonstration of a typical focal plane setup in which a non-polarizing beam splitter splits the beam into two. One branch is dumped while the other is reflected from a spatial light modulator before focusing onto the scattering sample, reprinted with permission [2].

Lensless setups do not experience the limitations from narrow fringe patterns because any modulation of the wavefront is directly projected onto the scattering sample itself. Thus, the ability to conjugate a scattering surface in the lensless setup is related to the SLM segment size and not fringe spacings. However, divergence of the beam now must be taken into account with the lensless setup but this problem can be mitigated by collimating the beam prior to modulation. As a result of removing the lens, the optimized spot produced in a lensless setup should therefore be more intense than the optimized spot produced in an image or focal plane system.

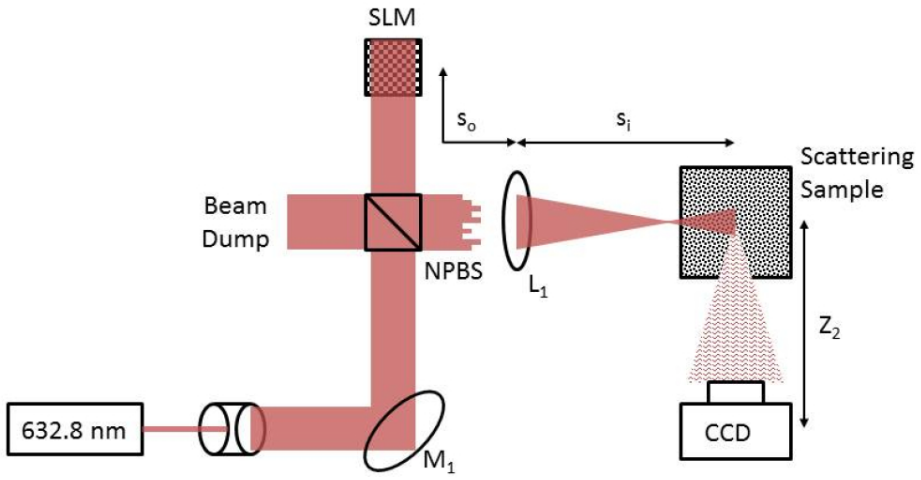


Figure 3. An image plane system differs from a focal plane system by creating an image of the SLM onto the scattering sample rather than focusing it through a lens, reprinted with permission [2].

2.3 Non-Mechanical Beam Steering

Once an appropriate phase mask or wavefront shape has been determined, light can be inversely diffused and made to focus after scattering from a sample. However, in order to utilize the principle of dual photography, several points of focus must be captured by the CCD at different spots in the scattering sample which would mean running through the algorithm to find an entirely new phase mask for each spot. Beam steering is one remedy to avoid having to create a new phase mask for each point of focus. Simply speaking, the easiest way to beam steer is to mechanically move the light source; however, to avoid tampering with the system in any way, non-mechanical beam steering could be achieved via the SLM. In order to create a path length difference that corresponds to a phase shift, a simple linear phase shift can be added to the optimized phase mask. If the phase shift is applied in a unique manner as to simulate a blazed grating, then the focused spot of light will shift to a different location depending upon the blaze angle. This result also demonstrates that

optimized phase masks for different locations are related to one another by a shift. Generally the blazed grating equation can be used to describe the steering angle as seen in

$$\sin(\theta) = \frac{m\lambda}{A}, \tag{1}$$

where θ is the steering angle, m is the order of steering, λ is the wavelength of light and A is the period of the grating [9]. This method of nonmechanical beam steering can shift energy into different orders depending upon the blaze angle of the grating. The angle of steering, however, can only take on discrete values where the value of the reset in the sawtooth profile is equivalent to a multiple of the wavelength of light or a multiple of a 2π phase shift [9].

Phase Tilts.

It is thus possible to decompose wavefront masks into two separate parts: the phase front that conjugates the rough surface to allow for inverse diffusion and the phase tilt that simulates a blazed grating to non-mechanically beam steer the optimized spot [8]. For the lensless setup, modulation of the wavefront can be treated as a simple linear combination. The focal plane system operates in a different manner due to the presence of the lens which focuses the light onto the scattering sample. To understand how to use phase tilts in the focal plane system, it is necessary to understand the Fourier domain and how the lens acts like a Fourier transform. Introducing a phase tilt or shift in one domain will result in the linear translation in the Fourier domain [10]. Therefore, creating a phase tilt at the spatial light modulator will translate the optimized spot on the scattering sample invalidating the wavefront shape that focuses the light at that specific spot [8]. Thus for the focal plane system the approach must be reversed and a linear translation of the spatial light modulator

is applied which results in a phase shift in the Fourier domain. Previous analysis on the focal plane system simulated the translation of the spatial light modulator by taking an optimized mask and shifting its pixels down a row and replacing the empty pixels with the bottom-most row [8] in post processing. Although spot displacement was achieved, it comes at the cost of the intensity of the optimized spot as some of the energy is being used to focus elsewhere on the scattering sample [8]. By removing the lens entirely, the optical system foregoes the use of the fourier transform in the implementation of phase tilts and beam steering and would theoretically obtain a more intense optimized spot since all pixels would be contributing [2].

2.4 Spatial Light Modulators

Spatial light modulators (SLM) are optical instruments that can modulate the phase or amplitude of light based upon some spatial pattern. SLMs are used in applications in plenty of optical fields such as beam steering, optical tweezers, diffractive optics, and pulse shaping among many more. Although the operation of SLMs varies, the SLMs used in this research are known as Liquid Crystal on Silicon (LCoS) SLMs and they rely on the birefringent nature of nematic liquid crystals to achieve modulation of light. Each pixel of the LCoS SLM contains liquid crystals whose orientation can be rotated through the application of a voltage that creates an electric field. Figure 4 demonstrates how the rotation of the liquid crystal affects the phase modulation by adjusting the extraordinary index of refraction. By adjusting the discrete values of the voltage, it is possible to modulate the phase of some wave after traveling through the liquid crystal layer of the SLM. The SLM can thus be thought of as a half-wave plate whose slow axis can be adjusted electronically [11]. It is therefore necessary that the incident light on the SLM be linearly polarized in the direction of the slow-axis for phase modulation to be observed.

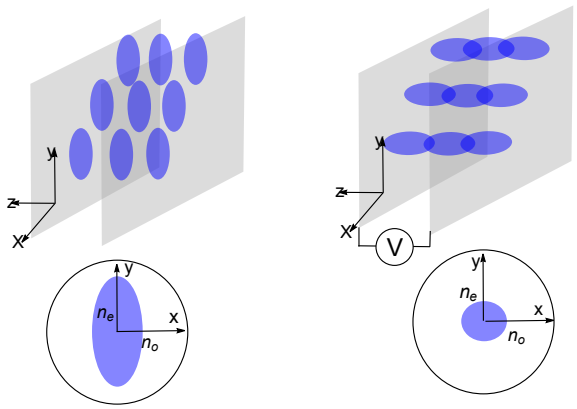


Figure 4. A demonstration of the birefringent nature of liquid crystals and how they are exploited in LCoS SLMs to achieve phase modulation recreated from [12]. A wavefront traveling on the z-axis will experience a large extraordinary index of refraction on the y-axis without any voltage as seen on the left. The application of a voltage as seen on the right will rotate the liquid crystal reducing the extraordinary index of refraction while leaving the ordinary index of refraction unchanged in both cases.

2.5 Wavefront Shaping

Wavefront shaping begins first by splitting an LCoS SLM into N segments to shape a wavefront before scattering it off some sample. The speckle pattern produced after scattering is then collected and characterized by identifying the reflected field, E, at the m^{th} position as seen in

$$E_m = \sum_{n=1}^{N} t_{mn} A_n e^{j\phi_n}, \tag{2}$$

where t_{mn} represents the m x n transport matrix associated with the scattering surface and $A_n e^{j\phi_n}$ is the field after having its phase modulated by the n^{th} segment of the SLM [6]. This transport matrix can be represented with any linear light transport process such as reflection without any change necessary in Equation (2) [7]. The intensity, I, at position, m, following scattering can be found after normalizing $A_n = \frac{1}{\sqrt{N}}$ as

$$I_m = |E_m|^2 = \frac{1}{N}\left|\sum_{n=1}^{N} t_{mn} e^{j\phi_n}\right|^2 . \tag{3}$$

The intensity of the speckle pattern in Equation (3) is defined as the metric for determining how to shape the incident field E to maximize transmission/reflection [7]. This metric is the enhancement, η, and it is given by

$$\eta = \frac{\langle I_{opt} \rangle}{\langle I_{rnd} \rangle}, \tag{4}$$

which is the ratio of the average intensity of the optimized spot to the average intesity of the background speckle pattern.

The maximum theoretical value of the intensity enhancement is related to the number of segments that describe the incident wavefront [4]. Equation (4) is used to derive the expression for the maximum enhancement in reflective inverse diffusion. In the reflective case, approximations have been made to give the scattering surface constant average reflectivity and phase delays that are related to the surface height fluctuations [2]. These approximations change Equation (3) into

$$I_m = |E_m|^2 = \frac{1}{N}\left|\sum_{n=1}^{N} \mathbf{r} e^{j\theta_{mn}} e^{j\phi_n}\right|^2 . \tag{5}$$

where $\mathbf{r}$ is the average surface reflectivity and θ_{mn} is the phase delay due to the surface height fluctuations [2]. For the optimized spot, the intensity is maximized when the phase delay from the SLM cancels out the phase delays from the surface height fluctuations, $\phi_n = -\theta_{mn}$. Thus Equation (5) becomes

$$\langle I_{max} \rangle = \left\langle \frac{1}{N}\left|\sum_{n=1}^{N} \mathbf{r}\right|^2 \right\rangle = \mathbf{r}^2 N \tag{6}$$

To determine the average intensity of the background speckle, the background is treated as a fixed-length random phasor sum which leads to

$$\langle I_{rnd} \rangle = \left\langle \frac{1}{N} \left| \sum_{n=1}^{N} \mathbf{r} e^{j\theta_{mn}} \right|^2 \right\rangle = \mathbf{r}^2 \left\langle \frac{1}{\sqrt{N}} \left| \sum_{n=1}^{N} e^{j\theta_{mn}} \right|^2 \right\rangle \tag{7}$$

Previous studies and reference material demonstrated that the ensemble average term in Equation (7) can be approximated to unity [2, 13]. Thus combining Equations 4, 6, and 7 provides the expression for the maximum theoretical value of enhancement in reflective inverse diffusion:

$$\eta_{reflective} = \frac{\langle I_{max} \rangle}{\langle I_{rnd} \rangle} = \frac{\mathbf{r}^2 N}{\mathbf{r}^2} = N \tag{8}$$

Equation (8) states that the more segments that can shape the incident wavefront, the higher the amount of energy that can be focused onto the target. It is not possible to know *a priori* the wavefront shape that will produce the largest enhancement, η. Algorithms were developed with the purpose of iteratively seeking out the best wavefront shape or phase mask that produces a maximum enhancement at a single point based upon feedback from the system itself. In practice, laboratory conditions limit the maximum theoretical enhancement from a wavefront mask due to decorrelation [14]. Decorrelation represents the temporal stability of the wavefront mask and depends upon the environment and the sample used. For example, if trying to image through living tissue the decorrelation time would be on the scale of seconds whereas a static sample could have a decorrelation time of several hours [14].

Feedback based Algorithms.

There are three main feedback based iterative algorithms that maximize the enhancement: the stepwise sequential algorithm, the partitioning algorithm, and the genetic algorithm [7].

The stepwise sequential algorithm or continuous sequential algorithm begins by

splitting the SLM into segments whose phase can be modulated from 0 to 2π. Each segment then modulates its phase which affects the intensity captured at the observation plane with a charged coupled device (CCD) camera. After iterating through all phase levels for that particular segment, the algorithm selects the phase modulation which resulted in the highest enhancement before moving onto the next segment as outlined in Figure 5.

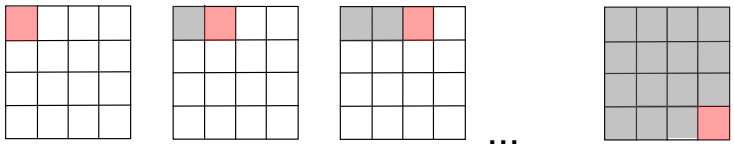


Figure 5. An example of how the stepwise sequential algorithm operates. The object represented in this figure is the segmented SLM. The stepwise sequential algorithm runs through individual segments of the SLM modulating the phase until the highest enhancement is achieved at a particular point of interest. After selecting the phase modulation with the highest enhancement produced, the algorithm moves sequentially onto the next segment repeating the process until the entire SLM has been optimized for the highest enhancement. The pink boxes represent pixels that are being modulated while grey boxes are the pixels that have been modulated and optimized.

The partitioning algorithm operates in a similar manner; however, rather than stepping through each segment piece by piece, the partitioning algorithm groups the SLM segments into partitions and modulates their phase as a whole. To accomplish this, half of the segments are chosen at random to have their phase modulated until the highest enhancement is achieved. After selecting the optimal phase modulations for that half of the SLM, a new half of the segments is chosen at random with a chance for previously optimized segments to be chosen yet again. This process is repeated until a satisfactory enhancement has been produced after scattering from the sample as seen in Figure 6. The largest obstacle faced when using the continuous sequential algorithm and the partitioning algorithm is the diminishing returns as time goes on [7]. The enhancement will increase in the region of interest greatly in the early stages

of the algorithm but will plateau out after the algorithm has been running for a period of time.

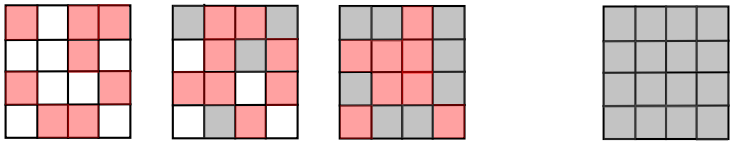


Figure 6. A figure detailing how the partitioning algorithm operates with the same color scheme as Figure 5. Half of the segments of the SLM are chosen at random to have their phase modulated simultaneously seeking the modulation that produces the largest enhancement. Once the optimal value has been selected, half of the segments are chosen once more which may or may not include previously optimized values.

Although the genetic algorithm is an iterative process, this type of algorithm takes a more Darwinian approach to finding the optimal phase mask to apply to the SLM that produces the highest enhancement as seen in Figure 7. The algorithm generates a population of phase masks with segments that have had their phases randomly set [15]. Each member of the population has its enhancement measured and is then stratified. Two parents are chosen to "breed" and create a new member to add to the population. Parents with a higher enhancement value are more likely to be chosen to breed and pass on their "qualities" that lend toward an improved enhancement. Each new offspring is then "mutated" by having a predetermined number of segments randomly modified. Once a new generation of phase masks has been produced, members of the population whose enhancement values are low are replaced by the newer generation with improved enhancement scores. This "evolution" process is repeated generation after generation until the desired enhancement has been produced at the point of interest. As time goes on and the population becomes more fit, the number of mutations is reduced to prevent any large scale changes near the end of the algorithm. The genetic algorithm has been demonstrated to be robust to high noise environments whereas the partitioning algorithm and the continuous sequential algo-

rithm are inhibited severely by noise [15]. However, the long calculation time of the genetic algorithm prevents its application in any dynamic media. Precaution should taken when selecting the initial parameters of the genetic algorithm as it is susceptible to getting stuck in local maxima [16]. The implementation of this algorithm in the lensless setup is discussed in further detail in Section 3.2.

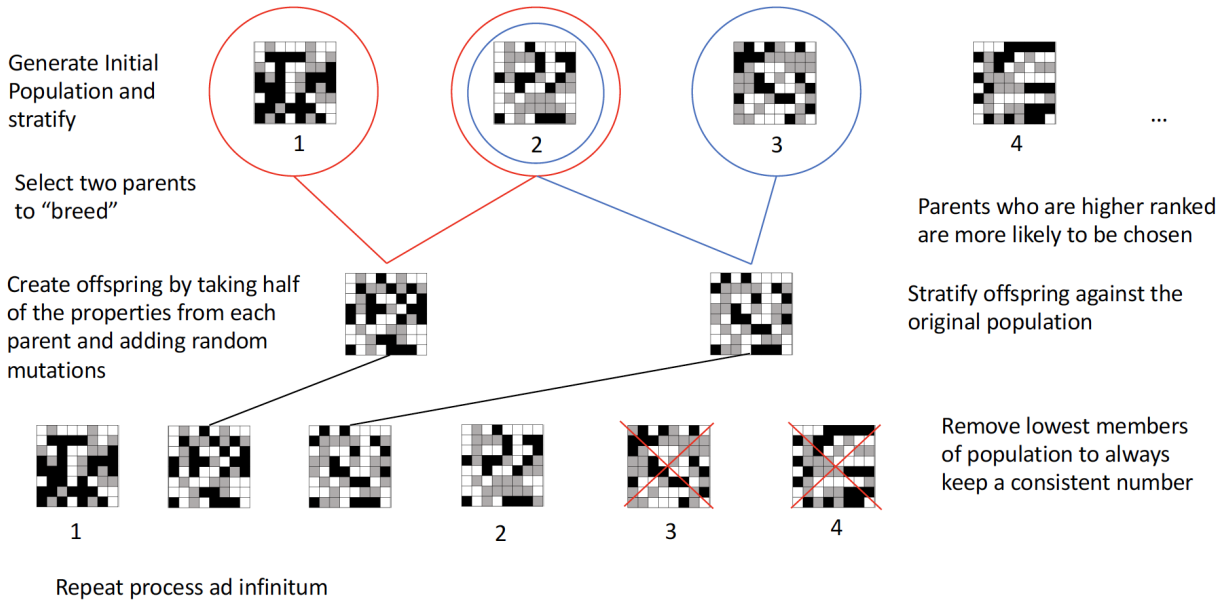


Figure 7. A description of the genetic algorithm used to find the wavefront mask that will produce the best enhancement.

III. Experimental Setup and Data Collection

3.1 Calibration of the SLM

The LCoS SLM used in this research is the Meadowlark Optics 1920 by 1152 XY Phase Series Spatial Light Modulator with Peripheral Component Interconnect Express (PCIe) controller. Calibration for this particular SLM comes in the form of a look up table (LUT) . Each pixel of the SLM can be modulated to 4096 different voltage values; however, not all voltage values will result in discrete and resolvable phase modulations. In addition, there is a non-linear relationship between the phase modulation and the voltage applied thus a LUT is required to operate the SLM which selects 256 voltages that correspond to linear phase modulations from 0 to 2π. In essence, the SLM can be thought of as a digital to analog converter which accepts an 8-bit digital input and translates it to an analog response via the LUT. Despite being provided a LUT for the SLM from the manufacturer, it is considered better practice to generate a custom LUT for individual use. The LUT is very sensitive to the conditions and environment of the SLM thus any deviations in the wavelength of light used or operating temperature of the SLM will render the manufacturer's LUT useless as it was created under specific conditions. To create a custom LUT, a Michelson interferometer setup is implemented in the initial calibration of the SLM as seen in Figure 8. The laser used in this experiment is a ThorLabs HRS015B Stabilized Helium Neon Laser with a wavelength of 632.991 nm. The beam is already linearly polarized in the vertical direction to be parallel to the slow axis of the SLM and it has been expanded prior to hitting a non-polarizing beamsplitter. The beam then splits and one leg reflects off of a fixed mirror while the other leg reflects off of the SLM before rejoining and combining where a CCD captures the interference fringe patterns.

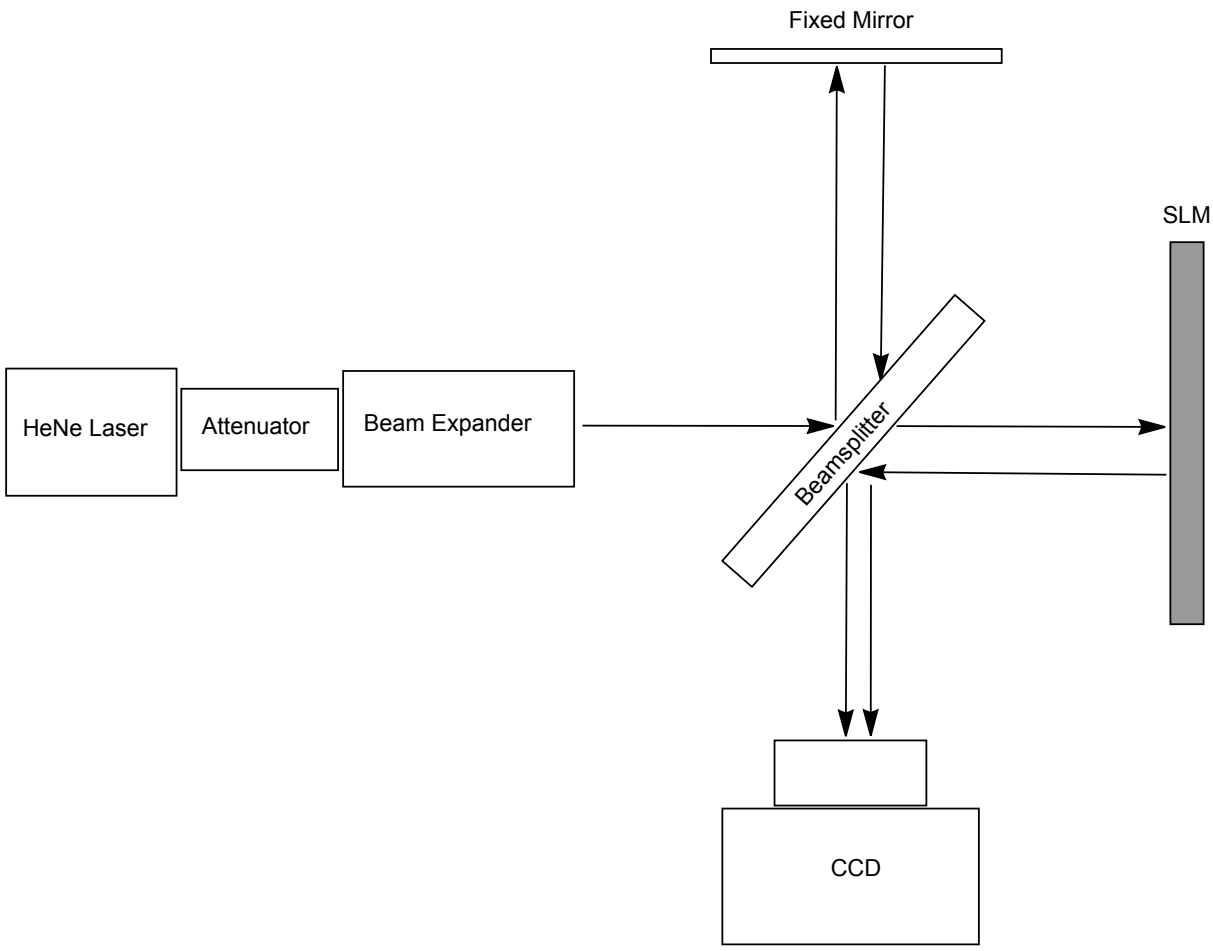


Figure 8. A conventional Michelson Interferometer setup in which the movable mirror is replaced by the SLM in order to generate a look up table that picks out 256 discrete voltage values that represent a linear phase shift from 0 to 2π.

Typically in a Michelson inteferometer experiment, the translation of the mirrors results in optical path length differences that manifest in the movement of the fringe patterns but in this case, the phase modulation of the SLM will result in movement of the fringes. To ensure that accurate phase measurements can be made, the fixed mirror was tilted to produce fringes of equal thickness as opposed to fringes of equal inclination as seen in Figure 9.

The SLM cannot function without a LUT thus an initial linear look up table is implemented for the purposes of calibration. The initial linear LUT is simply a function that takes the digital inputs from 0 to 255 and scales each input by a factor

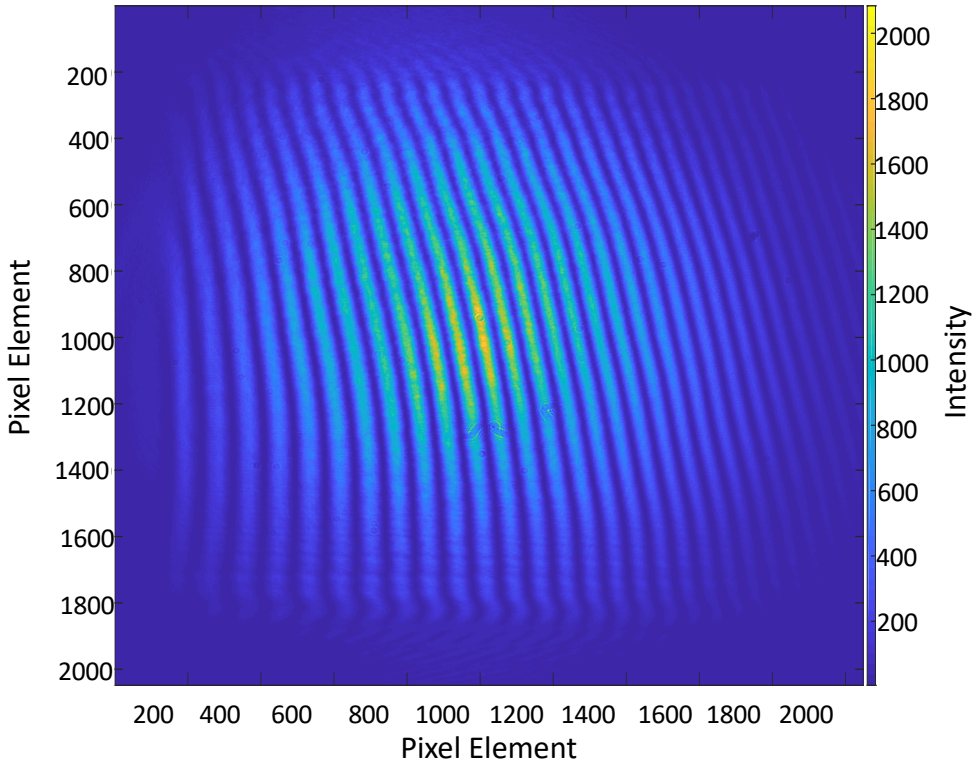


Figure 9. A demonstration of fringes of equal thickness that were obtained from the Michelson Interferometer calibration setup.

of 8.

The calibration process begins by assigning each one of the 1920 by 1152 pixels of the SLM a digital input of 0 to be used a reference. A single-row region of interest is applied to the inteferogram to capture how the intensity changes as a function of distance. Once the reference interferogram has been created, a new digital input is applied to each pixel of the SLM to generate a new interferogram. The resulting inteferogram should differ slightly from the reference and the phase difference between the two interferograms can be measured. The cross-correlation function is utilized to calculate the phase difference between the two interference patterns, which deter-

mines the similarity between two functions by overlaying the functions and sliding one of the functions while integrating. The location of the maximum value of the cross-correlation reveals the displacement between the two interferograms. To relate this length displacement to a phase displacement, the interference patterns are subjected to the Fourier Transform to calculate the wavelength, λ, of the fringe patterns. A simple proportional relationship can then be applied to relate the physical displacement, x, to the phase, ϕ as seen in

$$\frac{\lambda}{2\pi} = \frac{x}{\phi}. \tag{9}$$

After calculating the phase difference for this particular digital input, the entire process is repeated until the phase difference is calculated for all digital values 1-255. After several trials, it is possible to select certain values of the digital input so as to create a linear phase shift from 0 to 2π which represents the custom LUT for this particular experiment.

3.2 Experimental Setup

The setup for the experiment can be seen in Figure 10. The setup is similar to a focal plane setup; however, the lens has been removed to analyze the efficacy of beamsteering without the Fourier transform and to verify whether the enhancement obtained is higher as theorized [2], [8]. In addition, this setup uses a large beam expander to ensure that the SLM is completely illuminated with collimated light that is linearly polarized. The divergence of the beam following this collimation is approximately 0.035 mrad.

It is necessary to ensure that the entire array of pixels on the SLM is illuminated in order to guarantee that all segments contribute to the enhancement captured by the CCD after scattering. It is also important that the CCD be placed in the specular

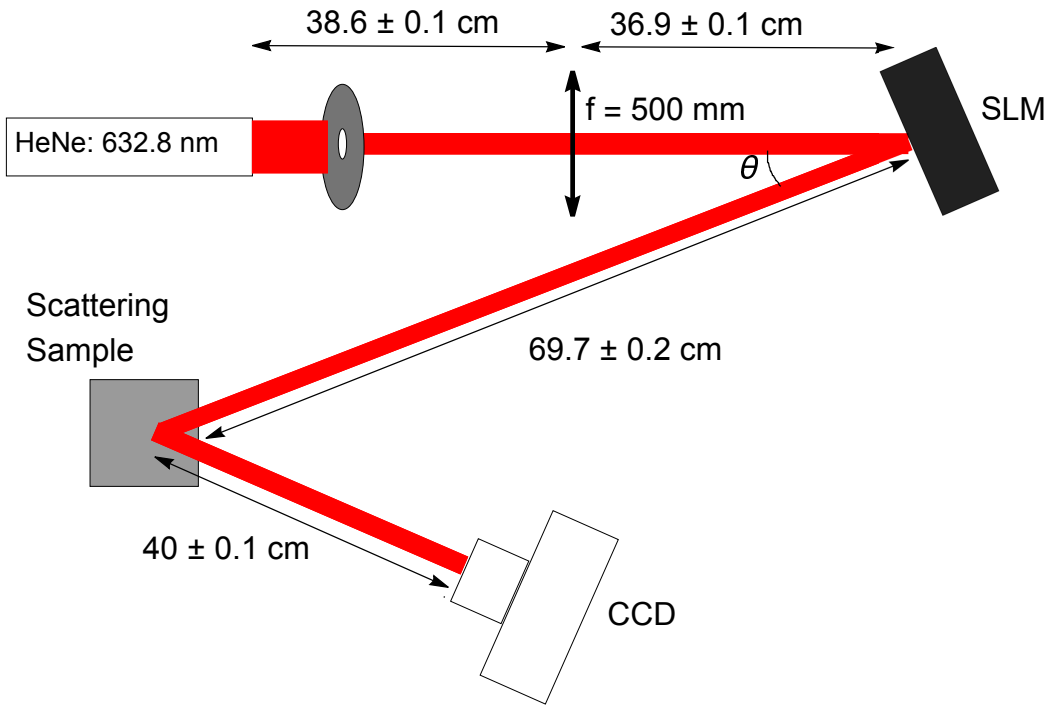


Figure 10. Experimental setup for lensless reflective inverse diffusion. In order to expand the beam, a pinhole spatial filter was placed directly in front of a linearly polarized helium neon laser. Afterwards, a lens with a fast focus is placed in front of spatially filtered light so that the beam is expanded to cover the entire SLM array with a collimated beam. The SLM is set an angle, $\theta = 12.6 \pm 0.1°$ where the light will reflect off the scattering sample. The CCD was placed in the specular region of the reflection off of the scattering sample.

region of the scattering sample to ensure a greater amount of energy is captured by the CCD. For beam steering, the scattering sample was oriented with an angle of incidence of $45° \pm 1°$. The maximum theoretical enhancement, in Equation (8), was determined to be a function of the number of individually controlled SLM segments. In addition to splitting the SLM into larger segments, the same operation is applied to the CCD to improve enhancement as much as possible. The size of the channel on the CCD should be the same size as the diffraction limited spot after scattering from the sample. The diffraction limited spot size for the lensless system can be approximated as

$$r \approx \frac{\lambda l}{D}, \tag{10}$$

where r is the radius of the spot size, λ is the wavelength of light in air, l is the distance from the scattering sample to the CCD, and D is the size of the sample because in the ideal case, the entire sample has been illuminated with modulated light from the SLM and is the controlling aperture. Equation (10) states that to maximize the enhancement, the channel spot size should be as large as possible which means making the segment sizes of the SLM as small as possible; however, the genetic algorithm runs too slowly to optimize the entire SLM pixel by pixel. Thus the SLM segment sizes were selected to be four pixels by four pixels which allowed the algorithm to run through 1364 generations in approximately 2.5 hours. The rationale for choosing 1364 generations was based upon previous studies of other iterative algorithms which determined that 1364 generations would give the same number of intensity measurements as other iterative algorithm experiments so that a direct comparison could be made between the focal plane system and the lensless system used in this experiment [8]. The dimensions of the spot size were calculated to be 14.4 microns by 23.6 microns using Equation (10) , thus the channel size of the CCD was set to be a rectangular shape of two pixels tall by three pixels wide.

Data Collection.

Data collection begins by utilizing the genetic algorithm previously outlined in Chapter 2. Initially, 30 members of the population of phase masks are created randomly and are ranked according to the enhancement of their speckle pattern produced at the region of interest which in this case is the center of the CCD. Images taken from the CCD have been rescaled in size to accomodate the size of the SLM with its 1920 pixels by 1152 pixels. The process to generate each random wavefront in

the initial population begins with creating a smaller mask that is 480 pixels by 288 pixels of phase values. From there, the smaller mask is scaled up by a factor of four to be 1920 pixels by 1152 pixels guaranteeing that each segment of the SLM is four pixels by four pixels. The enhancements of each wavefront mask in the population are calculated via equation (4) where the region of interest is $\langle I_{opt} \rangle$ and the background without the region of interest is $\langle I_{rnd} \rangle$.

Two parents are chosen to breed based on a linear weighting scheme, with a higher probability of being selected provided to members of the populations with high enhancement values. The breeding process begins by creating a random mask template of binary values that correspond to either the 'mother' or the 'father'. This random template is applicable for one pair of parents and is filled with elements taken from the halves of each parent in the population. After filling the template and producing the offspring, random mutations are then inserted into each offspring. These mutations eventually decay as the algorithm progresses as follows:

$$R = (R_0 - R_{end})e^{\frac{-n}{d}} + R_{end}, \tag{11}$$

where R is the rate of mutation, $R_0 = 0.1$ is the initial mutation rate, $R_{end} = 0.01$ is the final mutation rate, n is the generation number, and $d = 1364/3$ is the decay factor. The scattering sample utilized in this experiment is a 600 grit polished nickel sample. The HeNe laser was allowed to warm up for an hour prior to data collection to avoid any oscillations in power output.

Oversaturation.

Certain precautions have been implemented in order to ensure that the CCD is not oversaturated which causes the enhancement to plateau. This result is because after each generation, energy will be shifted towards the optimized spot by the SLM;

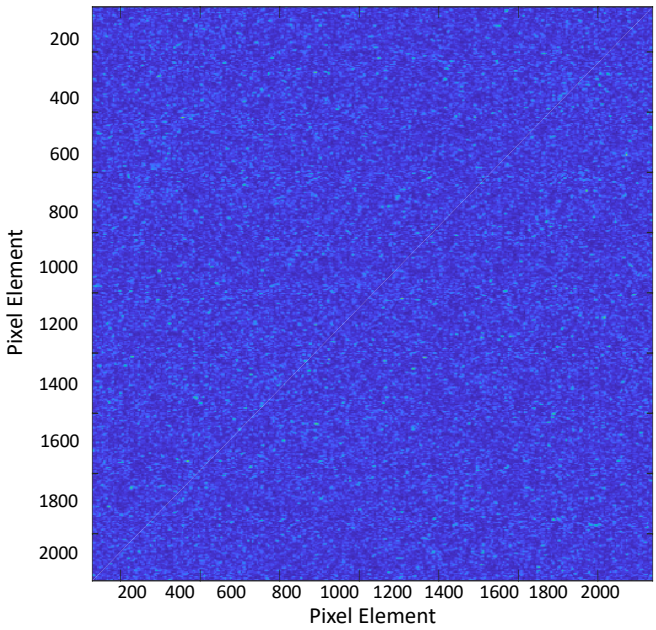
however, if the optimized spot has already reached the maximum intensity allowed by the CCD then further optimization of the phase mask will not be distinguishable and the enhancement will plateau. To prevent oversaturation and improve the enhancement obtained, a 0.5 OD attenuator is placed on the laser and the exposure time of the CCD is set to ten milliseconds. The attenuator will globally reduce the intensity of the laser when scattering off of the reflected sample; however, the increased exposure time of the CCD should allow the optimized spot to get brighter against the darker background without reaching the maximum intensity.

3.3 Experimental Results and Analysis

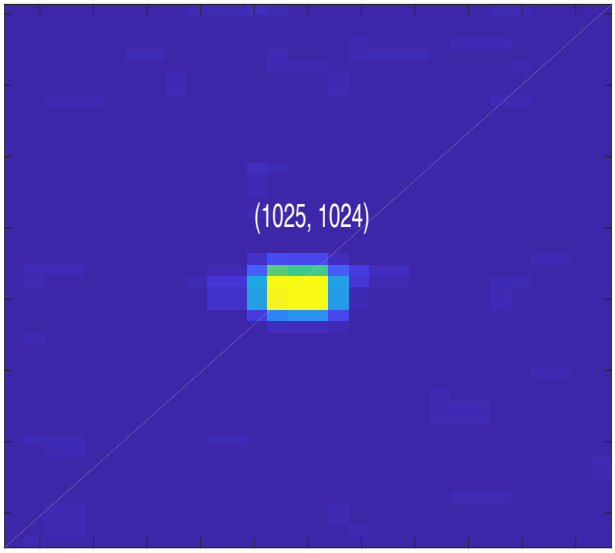
The results obtained from the genetic algorithm can be found in Figure 11. Figure 11 provides an example of a typical speckle pattern captured by the CCD before the genetic algorithm has run and the finished product after running through 1364 generations of the genetic algorithm.

Figure 12 demonstrates the average enhancement obtained over five trials as the genetic algorithm runs through 1364 generations. This is an increase in the enhancement from the focal plane setup using the same 600 grit piece of nickel which had an enhancement, $\eta \approx 89$ on average [8].

The uncertainty for these data sets is difficult to quantify as five trials is not sufficient to determine a limit ceiling or floor on the average enhancement over time. The enhancement of the optimized spot obtained in these laboratory conditions greatly differs from the theoretical enhancement found in Equation (8). Decorrelation plays a slight role in the reduction of enhancement over time as laboratory conditions prevent the enhancement from reaching values obtained in the simulations of reflective inverse diffusion [2] as seen in Figure 13. However, the decrease in the enhancement occurs very slowly as evident in Figure 13 which demonstrates that the enhancement of a



(a) 11a. Speckle Pattern



(b) 11b. Optimized Spot zoomed in

Figure 11. 11A, an example of a typical speckle pattern captured by the CCD after scattering from a 600 grit nickle sample with a exposure time of 20 milliseconds. The genetic algorithm then finds the optimized wavefront mask that will produce a focused spot at the center of the CCD that is many times brighter than the background as seen in 11b.

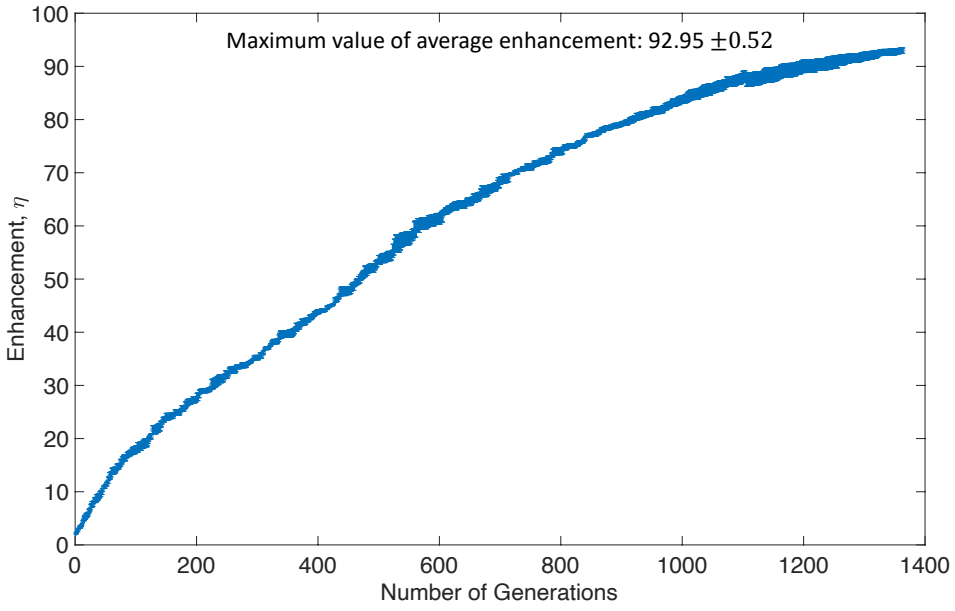


Figure 12. A plot of the average enhancement measured through each generation over five trial runs. Each of the five trial runs ran for 1364 generations and calculated a maxmium average enhancement of 92.95 ± 0.52. This data was collected under the same conditions as Figure 11.

particular mask decreased by a value of 2 over a period of 12 hours. Although the algorithm takes 2.5 hours to completely run through 1364 generations, decorrelation only plays a small role in the decay of the enhancement during data collection.

Potential factors that contribute to the limitation of the measured enhancement could be limitations in the equipment or the fact that genetic algorithm needs more time to reach theoretical values. For example, Figure 14 demonstrates the outcome when the genetic algorithm is allowed to run for 2000 generations rather than 1364. When unrestrained, the algorithm was able to reach enhancement values of 110 as opposed to when it could only reach average max values of 90 otherwise. Based on the

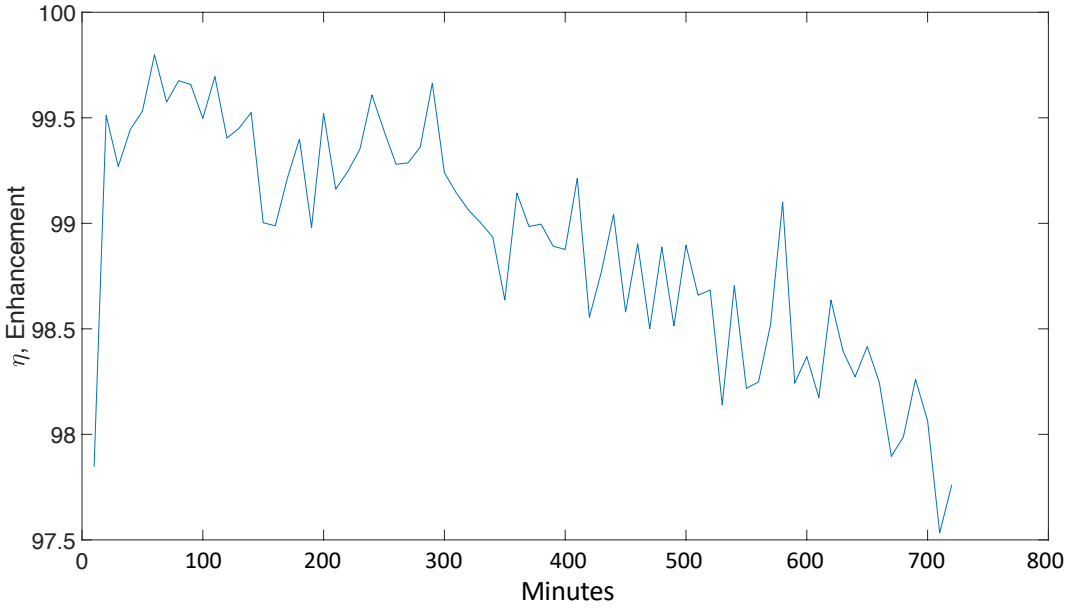


Figure 13. Over time, the wavefront mask that is optimized to produce a spot at some particular location will decay depending upon the sample and environmental conditions. This figure measures how the enhancement changes from a particular mask over a period of 12 hours collecting an observation point every ten minutes for a total of 72 data points.

average values obtained from Figure 12, the lensless system was able to outperform the focal plane system running under the same number of generations. To reach theoretical values of enhacement, the genetic algorithm would have to run for even longer and under conditions that limit the intensity of the background to ensure that the maximum intensity ceiling is not met by the CCD.

Beam Steering.

The main objective of this research isn't just to demonstrate that the enhancement has improved in the lensless system but to demonstrate that it can also utilize the

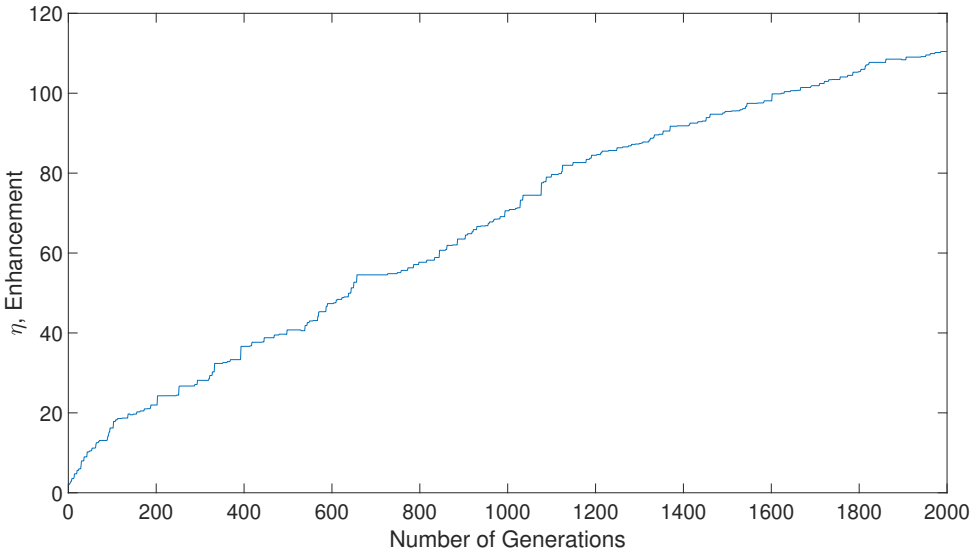
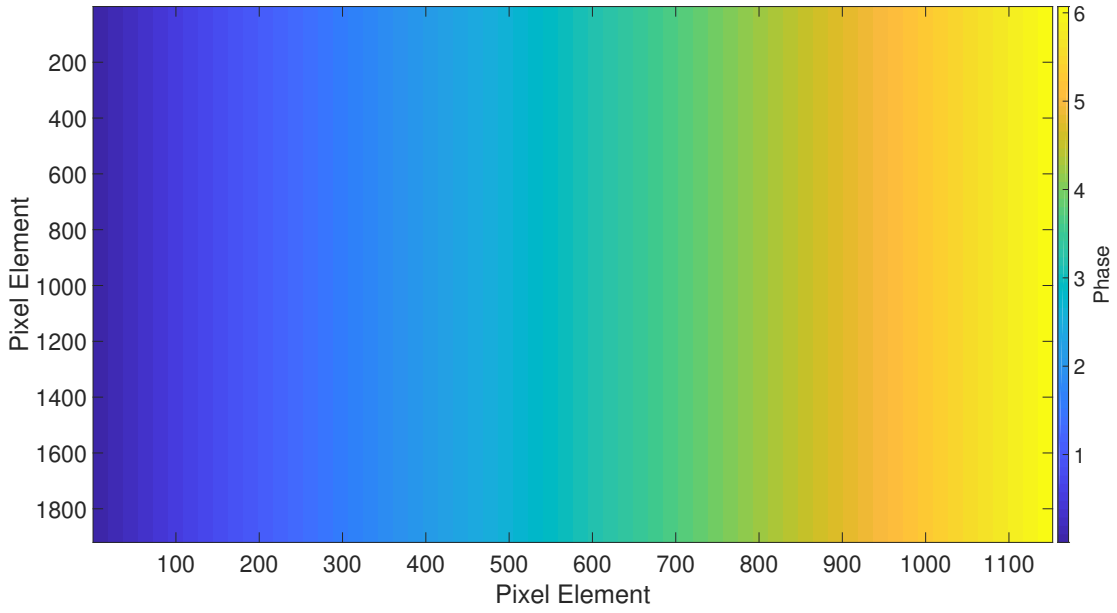


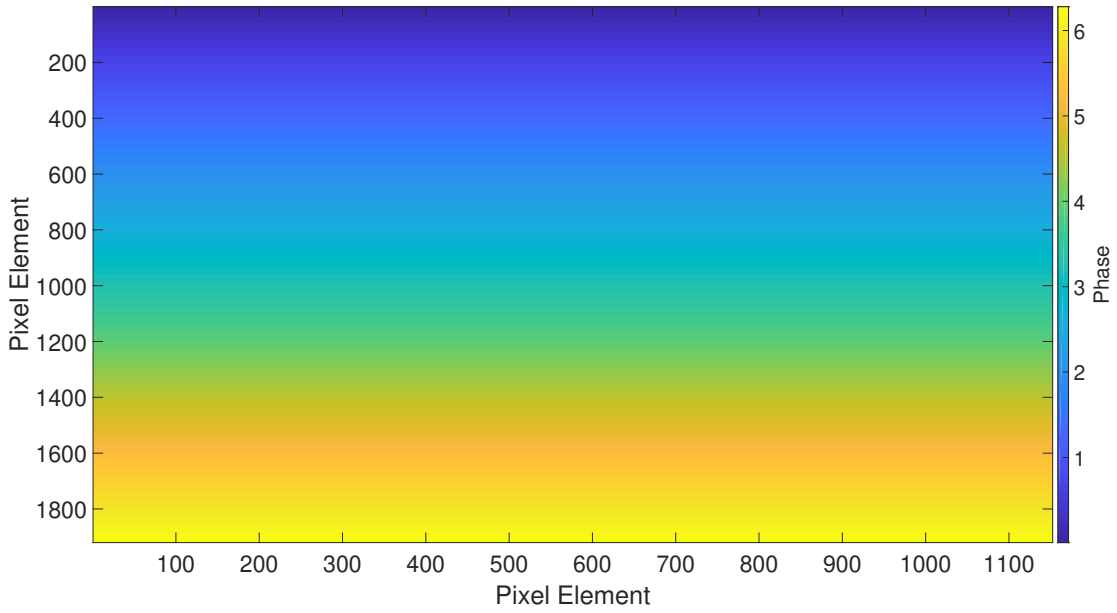
Figure 14. Using the same conditions in Figure 11, the genetic algorithm was allowed to run for 2000 generations to display how the enhancement is still climbing and hasn't yet reached a plateau. Setting the genetic algorithm to run for only 1364 was to allow for a direct comparison with the results obtained using the focal plane setup and to ensure that each trial would run for too long [8].

redundant phase information of neighboring solutions discovered in reflection matrices and steer the optimized spot [2]. To begin the process of beam steering, a blazed grating phase mask is first created. The grating takes interval steps from 0 to 2π along any dimension. This optimized spot will shift depending upon the orientation of the grating as evident in Figure 15.

Once an orientation has been chosen and the optimized wavefront mask has been identified by the genetic algorithm, the two masks can be linearly combined in order to impose a shift. To combine the optimized wavefront mask and the blazed grating mask, it is necessary to convert the wavefront mask from its integer values of 0 - 255 to the phase values (0 to 2π)using the LUT. After which, a simple addition along with a modulus will ensure that the two masks are combined properly to stay within the appropriate phase range. The sum of the masks must then be converted back into integer digital values which can be read by the SLM to implement the phase tilt



(a)

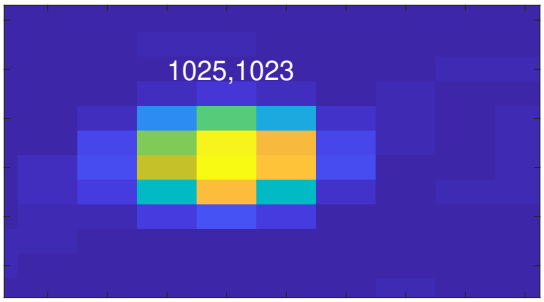


(b)

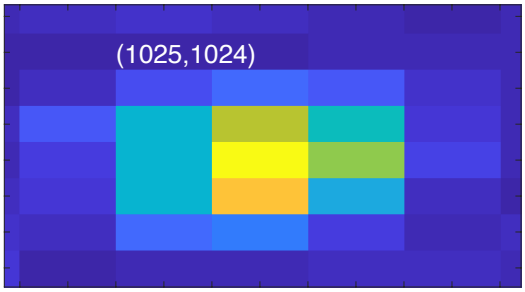
Figure 15. A series of blazed gratings which run from 0 to 2π in equal steps along a particular dimension to determine which direction to shift the optimized spot. A) will move the spot to the right. B) will shift the spot downwards.

and shift the optimized spot as in Figure 16. The different locations of the optimized spots in Figure 16 demonstrated successful beam steering as the optimized spot was able to shift upwards or downwards when the appropriate blazed grating was applied. The enhancement of each spot was greatly diminished once a shift was made. Each shift displaced the optimized spot by approximately a pixel on the CCD which is smaller than the sizes of the channels themselves. In addition, further applications of a particular blazed grating in a single direction should shift the optimized spot farther along that direction at the cost of enhancement of the original spot. This fact remains to be observed as any enhancement completely disappears on the second application of the blazed grating meaning that the region of interest and the background are indistinguishable from one other resulting in a speckle pattern.

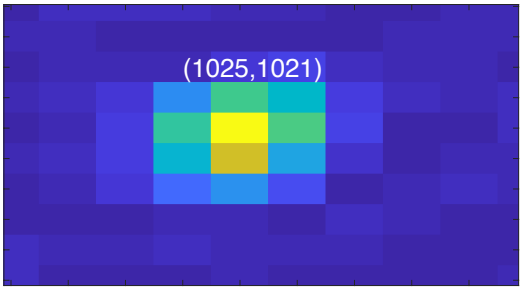
These results suggest that beam steering in the lensless system is not as effective as the focal plane system which conflicts with the results obtained from simulations [2, 8]. The distance that each optimized spot can shift on a single blazed grating is too small to be significant since each CCD channel is two pixels wide by three pixels tall and the displacement of the spots is on the order of a single pixel. In addition, the enhancement of the optimized spot is greatly reduced each time the blazed grating is combined with the optimized wavefront mask. One shift reduces the enhancement by approximately 77%, a reduction so significant that it severely limits trying to move the spot any further than a single pixel. Experimental results in the implementation of beam steering for the focal plane system saw a 50% reduction in the enhancement at approximately 0.23mm, nearly 31 pixels along the CCD [8]. A possible explanation for the poor performance of the lensless system in beam steering could be limitations in the LUT and SLM. The discrete values of the LUT from 0 to 2π may not be precise enough for minute phase differences needed for beam steering, which may also explain the large drops in enhancement over such a small displacement. Additionally,



(a)



(b)



(c)

Figure 16. A). The optimized spot following the genetic algorithm without any shifts with an enhancement of $\eta = 101.3 \pm 0.5$. B) The location of the optimized spot after applying an upward shift with enhancement, $\eta = 23.2 \pm 0.5$. C) The location of the optimized spot after applying an downward shift with an enhancement of $\eta = 26 \pm 0.5$. The location of each spot was determined by identifying the position of the top-left pixel of the channel.

the application of the grating phase map may actually contribute to steering the beam to another location on the diffuse sample which would mean that the pixels are no longer properly optimized to the original location on the sample driving down enhancement. Other limitations in the equipment could be aberrations in the system which wouldn't manifest during the genetic algorithm but during the implementation of beam steering, specifically during the conversion between integer and phase values using the LUT. These errors and abberations in the LUT conversion would be carried forward twice since there are two conversions when applying the phase grating to the SLM.

IV. Conclusion

4.1 Summary of Research

The primary objective of this research was to determine the efficacy of the lensless system against the focal plane system for achieving reflective inverse diffusion in multiple locations by exploiting phase information for non-mechanical beam steering. Inverse diffusion was achieved through the method of wavefront shaping: altering the incident plane wave so that the wave interacts with the diffuse surface and focuses rather than scattering in all directions.

This research implements the same iterative genetic algorithm that identified the optimized wavefront shape for the focal plane system. When the genetic algorithm was implemented in the lensless system, improvements in the enhancement of the optimized spot were observed. Although the lensless system outperformed the focal plane system, the lensless system did not prove itself to be effective in implementing phase tilts to beam steer the optimized spot. The data from this research suggests that despite the simplicity of the lensless system in the implementation of beam steering, the lensless system's effectiveness falls short of the focal plane system.

4.2 Future work and Extensions

Use of Spectralon as the scattering sample.

Spectralon is a type of material that exhibits extreme diffuse reflectance which is highly Lambertian. This type of surface behavior would more evenly distribute any scattering energy all around. In this experiment, special care was taken to ensure that the CCD was placed in the specular region of the scattering sample to focus as much energy as possible, but with spectralon the placement of the CCD wouldn't matter as much as it would be expected to be more uniform all around. Spectralon could also

allow for a larger steerable area which may help the loss of enhancement when trying to beam steer in the image plane system. Preliminary enhancement data obtained from using spectralon was significantly lower than the enhancement obtained from using the nickel sample. This could be due to the lambertian nature of spectralon prevents more energy from scattering in a particular direction like the nickel sample which has a specular region where more energy is scattered. Additionally, the lack of data which used spectralon as the scattering sample in the focal plane system prevented any direct comparison with the lensless system and therefore was not included in this analysis.

Improvements on the algorithm time.

Earlier it was mentioned that greater enhancement values of the optimized spot could be achieved by letting the genetic algorithm run for longer periods of time. When the algorithm ran for 1364 generations for a total period of time of approximately 2.5 hours, the enhancement reached an average value of approximately 92. Finding improvements or utilizing higher computational power may drastically cut down the algorithm time and achieve higher enhancements values faster for applicable use with dynamic systems.

Appendix A. Initializing Genetic Algorithm in Matlab ®

This code initializes the parameters needed to set up passing data to the SLM and taking observations from the CCD.

```
N=30; % Population size (was 30 in thesis)
G=15; % Number of offspring each generation (was 15 in thesis)

R0=.1; % Initial mutation rate - fraction of "mutated" segments in
    offspring
 % Initial mutation rate - fraction of "mutated" segments in offspring
Rend=.01; % Final Mutation rate
K=3000; % Max number of generations (use 10 for quick script check. Used
     1364 in thesis.)
Performance=zeros(1,K);

df=K/3; % Decay Factor
% Changing the number of SLM segments changes the diffraction limited
    spot
% size on the CCD... therefore the CCD dimension should be adjusted when
% the SLM dimension is adjusted

%slmpix=512; % old SLM: square 512x512. Used in fucntions.
slmpixc=1920; % width (cols) updated to new SLM - 1920x1152 rectangular
slmpixr=1152; % height (rows)0.0074

SLMrefresh=.18; % play with this (pauses)--long enough to modulate and
    camera to capture. (was 0.18)
%SLMrefresh=0; % no pause
```

```
%SLMdimension=128; % old=128--what should this be for 1920x1152?

%Split the SLM into channels as well, each channel is 4 by 4 pixels
   similar
%to CCD.

SLMdimensionc=480; % was 480 width (x) (ie. # cols)
SLMdimensionr=288; % was 288 height (y) (ie. # rows)

%SLMchannels=4096;
%SLMchannels=SLMdimension^2; % old=SLMdimension^2
SLMchannels=SLMdimensionr*SLMdimensionc;

CCDdimensionc=1024; % Channel size of CCD is 2 pixels wide by 3 pixels
   tall (spot size was determined to be 23.6 micron by 14.5 micron)
                       % spot size = (lambda * distance to CCD ) / (SLM
                           dimensions)

CCDdimensionr = 682; %2046 / 3

                       %This means that the full screen of the
                       %CCD: 2048 by 2048 will need to be cut
                       %short to make things an integer number.

CCDchannels=CCDdimensionc * CCDdimensionr; % 682 * 1024 = 698368

FrameAvg=1;

TestChannel = (CCDdimensionc ./ 2) .* (CCDdimensionr ./ 2); %center

[ChannelMask,BackgroundMask]=ChannelMasks(TestChannel,CCDdimensionc,
   CCDdimensionr, size(Iobs,1), size(Iobs,2)); % ChannelMask=zero
```

```
    matrix w/1's near center; BackgroundMask=1's matrix w/zeros near
    center

%Center Mask and Background Mask
ChannelMask = zeros(size(ChannelMask));

testmask = ones(3,2);
ChannelMask(1023:1025,1025:1026) = testmask;
ChannelMask = logical(ChannelMask);
BackgroundMask = ~ChannelMask; %Manually setting channel and background
    mask at the moment.
```

Appendix B. Generating Initial Population for initial stratification and performing iterations

```
Population_SLM = uint8(round(2^8 *rand(SLMdimensionr,SLMdimensionc,N),0)
    );

sfac = sqrt((slmpixc*slmpixr)/(SLMdimensionc*SLMdimensionr)); % new SLM
    scale factor: should be 4 (if channel size for SLM is 4 pixels by 4
    pixels)

 Population \_ SLM=imresize(Population_SLM,512/SLMdimension,'nearest'); %
     old SLM
Population \_ SLM=imresize(Population_SLM,sfac,'nearest');


% Calculate fitness of generation (BETH: or enhancement, of each mask in
     this generation of N masks)
Fitness=zeros(1,N);

for i=1:N

   fprintf('Loading mask %d \n', i);

   imageMatrix = flip(Population_SLM(:,:,i)');
   imageVector = typecast(imageMatrix(:), 'uint8');
   imageBuffer = libpointer('uint8Ptr', imageVector);
    calllib('Blink_C_wrapper','Write_image', 1, imageBuffer, slmpixc *
        slmpixr, 0, 0, 5000);

   calllib('Blink_C_wrapper','ImageWriteComplete',1, 5000);
```

```
    disp('ImageWriteComplete');
         % load mask
    pause(SLMrefresh); % SLM LC refresh 30 Hz
     tlCamera.FreeAllButGivenNumberOfFrames(0); % Clear the buffer to
         ensure no frames of un-modulated SLM are shown...
    imageData2D=getavgimage(tlCamera,FrameAvg);
           % get CCD image
    Iobs=(imageData2D'); % if an error shows up saying that imageData2D
       doesnt exist, it means the buffer isn't being filled in time
    Iobs(2047:2048,:) = [];

    clear imageData2D % we clear this variable to double check that we
       are getting a new frame every time

    % Choose how to define Fitness:
    %Fitness(i)=sqrt(mean(Iobs(ChannelMask).^2)); % Channel Intensity
    Fitness(i)=sqrt(mean(Iobs(ChannelMask).^2))/sqrt(mean(Iobs(
       BackgroundMask).^2)); % Channel Enhancement
    %Fitness(i)=sqrt(WeightedIntensity2(Iobs(ChannelMask),2048/binfactor/
       CCDdimension)); % Weighted Channel intensity
    %Fitness(i)=sqrt(WeightedIntensity2(Iobs(ChannelMask),2048/binfactor/
       CCDdimension))/sqrt(mean(Iobs(BackgroundMask).^2)); % Weighted
       Channel Enhancement--was using this one!
end

% Sort the SLM screens by fitness
[Fitness,fitorder]=sort(Fitness,'descend'); % sort fitness of each mask
    in this generation from high to low

Population_SLM(:,:,:)=Population_SLM(:,:,fitorder); % sort masks in this
    generation from best to worst
```

Iterations Begin.

```
graphics=false; % false=off, true=on

X=1;
ThresholdFitness=0;
%Used to measure fitness of offspring
Offspring_Fitness=zeros(1,G);

for n=1:K % n is the generation number

   fprintf('Generation %d \n', n);
% Offspring_SLM=zeros(1152,1920,G,'uint16'); % old SLM dimensions (row,
    cols)--was 512,512
   Offspring_SLM=zeros(1152,1920,G,'uint8'); % needs 8-bit input rather
       than 16 bit
   for i=1:G

      % Choose parents - Different weighting schemes for probability of
          a
      % parent being chosen

      %ParentIndices=randsample(N,2,true,(N:-1:1)); % linear Weights
      %ParentIndices=randsample(N,2,true,(N:-1:1).^1.5); % nonlinear
          Weights
      ParentIndices=randsample(N,2,true,(N:-1:1).^(X)); % variable
          nonlinear Weights
      %ParentIndices=randsample(N,2,true,Fitness.^2); % Fitness-based
          weights

      % Generate binary template
```

```
        Template=rand(SLMdimensionr,SLMdimensionc); % new SLM (old was
            512,512)
        Template=imresize(Template,sfac,'nearest'); % new SLM scale factor
             (old was 512/SLMdimension)
        Template=Template>.5;
        %Use the template to combine half the segments from one randomly
        %chosen parent with the other randomly chosen parent

        %Offspring_SLM(:,:,i)=Population_SLM(:,:,ParentIndices(1)).*uint16(
            Template)+Population_SLM(:,:,ParentIndices(2)).*uint16(1-
            Template);
         Offspring_SLM(:,:,i)=Population_SLM(:,:,ParentIndices(1)).*uint8(
             Template)+Population_SLM(:,:,ParentIndices(2)).*uint8(1-
             Template);
        % Generate mutation template
        Template=rand(SLMdimensionr,SLMdimensionc); %
        Template=imresize(Template,sfac,'nearest');
        Template=Template<(R0-Rend)*exp(-n/df)+Rend; % The term on the
            right hand side is the proportion of pixels to be mutated...
            see Conkey paper

% Mutation=uint16(round(2^16*rand(SLMdimensionr,SLMdimensionc),0)); %
    Random sample for old SLM
        Mutation=uint8(round(2^8*rand(SLMdimensionr,SLMdimensionc),0)); %
            New SLM requires 8-bit random mutation rather than 16-bit
        Mutation=imresize(Mutation,sfac,'nearest'); % scale factor updated

        % As before, use the template to replace a number of the
        % offspring's segments with the mutation
        %Offspring_SLM(:,:,i)=Offspring_SLM(:,:,i).*uint16(1-Template)+
            Mutation.*uint16(Template);
```

```
        Offspring_SLM(:,:,i)=Offspring_SLM(:,:,i).*uint8(1-Template)+
            Mutation.*uint8(Template);
    end

    % Measure fitness of each offspring by loading it into SLM

    for i=1:G

% Offspring_SLM = flip(Offspring_SLM);

    fprintf('Loading offspring %d \n',i);
    offimageMatrix = flip(Offspring_SLM(:,:,i)');
    offimageVector = typecast(offimageMatrix(:), 'uint8');
    offimageBuffer = libpointer('uint8Ptr', offimageVector);
    calllib('Blink_C_wrapper','Write_image', 1, offimageBuffer, slmpixc *
        slmpixr, 0, 0, 5000);

    calllib('Blink_C_wrapper','ImageWriteComplete',1, 5000);
    disp('offspring_ImageWriteComplete');
        pause(SLMrefresh); % SLM LC refresh 30 Hz
        tlCamera.FreeAllButGivenNumberOfFrames(0);

        imageData2D=getavgimage(tlCamera,FrameAvg);
    Iobs=(imageData2D'); % if an error shows up saying that imageData2D
        doesnt exist, it means the buffer isn't being filled in time
   Iobs(2047:2048,:) = []; % we clear this variable to make sure we are
       getting a new frame every time
   clear imageData2D
   %Offspring_Fitness(i)=sqrt(mean(Iobs(ChannelMask).^2)); % Channel
       Intensity
        Offspring_Fitness(i)=sqrt(mean(Iobs(ChannelMask).^2))/sqrt(mean(
            Iobs(BackgroundMask).^2)); % Channel Enhancement
```

```
        %Offspring_Fitness(i)=sqrt(WeightedIntensity2(Iobs(ChannelMask)
            ,2048/binfactor/CCDdimension));
        %Offspring_Fitness(i)=sqrt(WeightedIntensity2(Iobs(ChannelMask)
            ,2048/binfactor/CCDdimension))/sqrt(mean(Iobs(BackgroundMask)
            .^2)); % was using this one!
    end

    % Replace lowest ranked masks with new generation of G offspring,
        then rank
    Population_SLM(:,:,N-G+1:end)=Offspring_SLM;
    Fitness(N-G+1:end)=Offspring_Fitness;

    % Rank... Sort the SLM screens by fitness
    [Fitness,fitorder]=sort(Fitness,'descend');
    Population_SLM(:,:,:)=Population_SLM(:,:,fitorder);
% Fitness
% n
    if Fitness(N-G)==ThresholdFitness
        X=X+1;
    else
        X=1;
        ThresholdFitness=Fitness(N-G);
    end
    %X


    if X>5 % if number of iterations where the surviving fraction of the
         population does not change is greater than 20,
        % then, it is time to reassess the fitness of the SLMs
        for i=1:N
            newimageMatrix = flip(Population_SLM(:,:,i)');
            newimageVector = typecast(newimageMatrix(:), 'uint8');
```

```
      newimageBuffer = libpointer('uint8Ptr', newimageVector);
      calllib('Blink_C_wrapper','Write_image', 1, newimageBuffer,
          slmpixc * slmpixr, 0, 0, 5000);
      disp('Write_image')
      calllib('Blink_C_wrapper','ImageWriteComplete',1, 5000);
      disp('ImageWriteComplete'); %new
      pause(SLMrefresh); % SLM LC refresh 30 Hz
      tlCamera.FreeAllButGivenNumberOfFrames(0);

      imageData2D=getavgimage(tlCamera,FrameAvg);
 Iobs=(imageData2D'); % if an error shows up saying that imageData2D
     doesnt exist, it means the buffer isn't being filled in time
Iobs(2047:2048,:) = []; % we clear this variable to make sure we are
    getting a new frame every time
clear imageData2D
%Fitness(i)=sqrt(mean(Iobs(ChannelMask).^2)); % Channel Intensity
      Fitness(i)=sqrt(mean(Iobs(ChannelMask).^2))/sqrt(mean(Iobs(
          BackgroundMask).^2)); % Channel Enhancement
      %Fitness(i)=sqrt(WeightedIntensity2(Iobs(ChannelMask),2048/
          binfactor/CCDdimension));
      %Fitness(i)=sqrt(WeightedIntensity2(Iobs(ChannelMask),2048/
          binfactor/CCDdimension))/sqrt(mean(Iobs(BackgroundMask).^2)
          ); % was using this one
   end
   [Fitness,fitorder]=sort(Fitness,'descend');

X=0;
end

%%
Performance(n)=max(Fitness);
max(Fitness)
```

```
end
```

Appendix C. Creating Channel and background masks to focus on a particular channel

```
function [ChMask,BkMask]=ChannelMasks(TestChannel,CCDdimensionc,
    CCDdimensionr, m, n)

%[m,n]=size(Iobs);

%CCDdimension=sqrt(CCDchannels);

%TestChannel=90;
% column=ceil(TestChannel/CCDdimensionc);
% row=TestChannel-(column-1)*CCDdimensionr;

Spot=ones(3,2);
SmallChannelMask = zeros(CCDdimensionr, CCDdimensionc);
SmallChannelMask((CCDdimensionr / 2): ((CCDdimensionr / 2) + 2), (
    CCDdimensionc / 2):((CCDdimensionc/2) + 1)) = Spot;

%function 'Expand Image' will expand the small channel mask so that it
    fits
%over the dimensions of the CCD. The small channel mask will replace all
%values with 1 or 0 only.


ChMask=ExpImage(SmallChannelMask >=1, m,n); %Inserted small channel
    mask but only the value of 2??
BkMask=ExpImage(~SmallChannelMask,m,n);

end
```

```
% SmallChannelMask=zeros(CCDdimension);
% SmallChannelMask(TestChannel)=1;

% [row,column]=find(SmallChannelMask);
%
% if mod(row-1,CCDdimension)==0
% rstart=1;
% rstop=row+1;
% elseif mod(row,CCDdimension)==0
% rstart=row-1
% rstop=row;
% else

% if TestChannel==1;
% SmallChannelMask(TestChannel:(TestChannel+1))=1;
% SmallChannelMask((TestChannel+CCDdimension):(TestChannel+CCDdimension
   +1))=1;
% elseif TestChannel>1 && TestChannel<CCDdimension
% SmallChannelMask((TestChannel-1):(TestChannel+1))=1;
% SmallChannelMask((TestChannel+CCDdimension-1):(TestChannel+
   CCDdimension+1))=1;
% elseif TestChannel==CCDdimension
%
% elseif TestChannel>(CCDchannels-CCDdimension+1) && TestChannel<
   CCDchannels
%  SmallChannelMask((TestChannel-CCDdimension-1):(TestChannel-
   CCDdimension+1))=1;
% SmallChannelMask((TestChannel-1):(TestChannel+1))=1;
% elseif mod(TestChannel-1,CCDdimension)==0
% SmallChannelMask((TestChannel-1):(TestChannel+1))=1;
% elseif mod(TestChannel-1,CCDdimension)==0
% SmallChannelMask((TestChannel-1):(TestChannel+1))=1;
```

```
% end
```

Appendix D. Expanding images to scale

```
function [out] = ExpImage(frame,p,q)

%TESTING ONLY CENTER CHANNEL, NEED TO ADJUST CODE IF OTHER SECTIONS NEED
    TO
%BE TESTED
[m,n]=size(frame);

if nargin <= 3
   if length(p) == 1
      rows=p/m;
      if mod(rows,1)~=0
         error('not integer resize');
      end
      cols=q/n;
      if mod(q,1)~=0
         error('not integer resize');
      end
   else
      error('improper input');
   end
end

vx = ceil((1:m*rows)/rows);
vy = ceil((1:n*cols)/cols);

out=frame(vx,vy);

end
```

Appendix E. Capturing frames from the CCD and averaging them

```
function imageData2D=getavgimage(tlCamera,NumFrames)
imageData2D=getimage(tlCamera);
imageData2D(:,:,2:NumFrames)=zeros([size(imageData2D),NumFrames-1]);
for i=2:NumFrames
   imageData2D(:,:,i)=getimage(tlCamera);
end
imageData2D=mean(imageData2D,3);
end
```

Appendix F. Creating a blazed grating to shift optimized spot

```
function output = createShift(x,index,boolean)
slm = double(zeros(1920,1152));

center_c = 1920 / 2;
center_r = 1152 / 2;

% shift = uint8(zeros(108));

if index == 1
   if boolean == 1
      for i = 1:1:size(slm,1)
         slm(i,:) = mod((i * x),(2 * pi));
      end
   else
      for i = 1:1:size(slm,1)
          slm(i,:) =(2 * pi) - mod((i * x),(2 * pi));
      end
   end

elseif index == 2
   if boolean == 1
      for i = 1:1:size(slm,2)
         slm(:,i) = mod((i * x),(2 * pi));
      end
   else
      for i = 1:1:size(slm,2)
         slm(:,i) = (2 * pi) - mod((i * x),(2 * pi));
      end
```

```
    end
else
    error('Index must be 1 or 2');
end


output = slm;
end
```

Appendix G. Converting analog input to phase and vice versa

Digital input values to phase values .

```
function output = matrix2phase(inputmatrix)

[p,q] = size(inputmatrix);
output = double(zeros(p,q));
inputmatrix = double(inputmatrix);

for i = 1:1:p
   for j = 1:1:q
      output(i,j) = (inputmatrix(i,j) .* (2 * pi)) ./ 255; % should
         convert the integer values into radians and store them into
         output
   end
end

end
```

Phase values to digital input values.

```
function output = phase2matrix(inputphasemap)

[p,q] = size(inputphasemap);

output = uint8(zeros(p,q));

for i = 1:1:p
   for j = 1:1:q
```

```
        output(i,j) = inputphasemap(i,j) .* 255 ./ (2 * pi);
    end
end

end
```

Bibliography

1. P. Sen, B. Chen, G. Garg, S. R. Marschner, M. Horowitz, M. Levoy, and H. P. A. Lensch, "Dual photography," *ACM Trans. Graph.*, vol. 24, p. 745–755, July 2005.

2. K. W. Burgi, *Reflection Matrix Method for Controlling Light After Reflection From a Diffuse Scattering Surface*. PhD thesis, Air Force Institute Of Technology WRIGHT-PATTERSON AFB OH Wright-Patterson ..., 2016.

3. M. G. Hoelscher and M. A. Marciniak, "Restoration of scene information reflected from a non-specular surface," in *Reflection, Scattering, and Diffraction from Surfaces II* (Z.-H. Gu and L. M. Hanssen, eds.), vol. 7792, pp. 148 – 159, International Society for Optics and Photonics, SPIE, 2010.

4. I. M. Vellekoop and A. P. Mosk, "Focusing coherent light through opaque strongly scattering media," *Opt. Lett.*, vol. 32, pp. 2309–2311, Aug 2007.

5. I. Freund, "Looking through walls and around corners," *Physica A: Statistical Mechanics and its Applications*, vol. 168, no. 1, pp. 49 – 65, 1990.

6. I. Vellekoop and A. Mosk, "Focusing of light by random scattering," *arXiv preprint cond-mat/0604253*, 2006.

7. I. M. Vellekoop, "Feedback-based wavefront shaping," *Opt. Express*, vol. 23, pp. 12189–12206, May 2015.

8. E. Nagamine, "The non-mechanical beam steering of light in reflective inverse diffusion," Master's thesis, Air Force Institute Of Technology WRIGHT-PATTERSON AFB OH Wright-Patterson ..., 2019.

9. P. F. McManamon, P. J. Bos, M. J. Escuti, J. Heikenfeld, S. Serati, H. Xie, and E. A. Watson, "A review of phased array steering for narrow-band electrooptical systems," *Proceedings of the IEEE*, vol. 97, no. 6, pp. 1078–1096, 2009.

10. J. W. Goodman, "Introduction to fourier optics," 2005.

11. "User manual: 1920 x 1152 xy phase series spatial light modulator with pcie controller."

12. P. Balondrade and S. M. Popoff, "Calibration of Linearly Aligned Nematic Liquid Crystal based Spatial Light Modulators," tech. rep., Sept. 2018.

13. J. W. Goodman, *Speckle phenomena in optics: theory and applications*. Roberts and Company Publishers, 2007.

14. I. M. Vellekoop and A. Mosk, "Phase control algorithms for focusing light through turbid media," *Optics communications*, vol. 281, no. 11, pp. 3071–3080, 2008.

15. D. B. Conkey, A. N. Brown, A. M. Caravaca-Aguirre, and R. Piestun, "Genetic algorithm optimization for focusing through turbid media in noisy environments," *Opt. Express*, vol. 20, pp. 4840–4849, Feb 2012.

16. C. Gong, T. Wu, J. Liu, H. Li, X. Shao, and J. Zhang, "Focusing of light through turbid media by curve fitting optimization," *Optical Engineering*, vol. 55, no. 12, p. 123105, 2016.