

Acknowledgements

I would first like to thank the members of my examining committee, Professor Jan Chaloupka, Professor Gina Hoatson, and Professor George Greenia for all of their efforts on my behalf. A special thank you goes to my advisor Professor Jan Chaloupka for his wisdom, guidance, and support throughout the course of my research. I would also like to thank William and Mary graduate student Kelly Kluttz for sharing her knowledge of lasers and the SLM. Another thank you to Anna Linnenberger from Boulder Nonlinear Systems Inc. for her sound advice and swift response to all of my SLM related inquiries. Lastly, I would like to thank William and Mary senior William Abbott for his assistance in formatting and editing of this thesis.

Abstract

We describe the modification of the spatial amplitude profile of a laser beam in the far field through phase modulation of the laser beam in the near field. Utilizing a computer controlled elitist genetic algorithm technique we are able to program a Spatial Light Modulator (SLM) to modify the phase of the beam on a pixel by pixel basis. Without external input from the user, the algorithm modifies the phase of the beam until the shape of the focal spot satisfies a predefined fitness criterion. These techniques have many interesting experimental applications including adaptive optics schemes and phase control of processes driven by a femtosecond laser.

Contents

1. Introduction and Background	3
1.1 Fourier Optics	3
1.2 Phase Modulation	6
2. Experimental	8
2.1 Experimental Setup	8
2.2 Spatial Light Modulator	10
3. Genetic Algorithm	14
3.1 Theory	14
3.2 Laser Application	15
4. Results	20
4.1 Pre-Load Generic	20
4.2 Pre-Load Block	23
4.3 Draw Block	27
5. Conclusion	29

1. Introduction and Background

With the rapid adoption of lasers as the light source of choice in the optics community, an increasing number of experiments have been conducted focused on manipulating the shape of laser pulses [1,2]. A large number of these experiments have also involved the utilization of a Spatial Light Modulator (SLM) to alter light on a pixel-by-pixel basis, and change both the spatial intensity and phase distribution of the laser beam. For example, phase modulation of a laser beam using a programmable SLM has been used to generate hollow laser beams [3].

We chose to focus on phase modulation of a laser source using a programmable SLM to vary the phase of the laser beam. This required a tabletop optical setup consisting of a helium-neon laser, a SLM, and a Charged Coupled Device (CCD) camera. We controlled our setup using a “survival of the fittest” computer program more commonly known as a genetic algorithm. The algorithm instructs the SLM to iterate the phase of the laser pulse thereby modifying the spatial profile of the beam. These modifications are detected by the CCD camera and the program runs until a specified focal spot image is achieved. The applicability of this feedback approach is widespread and has been demonstrated in previous experiments [4]. Other studies utilize a feedback approach that looks at temporal modifications of a laser beam [5]. However, in contrast our work focuses on spatial modification of a laser beam.

1.1 Fourier Optics

Fourier analysis, developed by French Mathematician Joseph Fourier in 1822, is the cornerstone of several vital theories in mathematics as well as in physics. Fourier’s

theory holds that any function can be broken down into its harmonic components according to the following equation (specified in one dimension),

$$f(x) = 1/\pi[\int_0^\infty A(k) \cos(kx) dk + \int_0^\infty B(k) \sin(kx) dk]. \quad (1)$$

The harmonic contributions of the Fourier equation (namely the $A(k)$ and $B(k)$ coefficients) form the respective real and imaginary components of the “Fourier Transform;” that is, a new function which describes the spatial frequency of the input spatial function ($f(x)$). After some manipulation this relation is stated as follows:

$$F(k) = \int_{-\infty}^\infty f(x)e^{ikx} dx. \quad (2)$$

This equation represents a mapping from the spatial realm (x) to frequency space (k).

Physicists use Fourier analysis for a broad range of applications, however of most interest to our research is the field of Fourier Optics, simply defined as the way in which optical systems modify light to form images [6]. An important feature of almost all optical systems is that the optical pattern through a lens at the focal plane corresponds exactly to the Fourier Transform of the input light source. Of course the transform associated with a lens is in two dimensions, so an appropriate transform relation is needed to calculate the function in frequency space. Following the model in Goodman, the equation below represents the intensity distribution in the focal plane of a thin lens, with $t_A(x, y)$ describing the amplitude transmittance of a monochromatic plane wave of amplitude A [7].

$$I_f(u, v) = (A^2/\lambda^2 f^2)[\iint_{-\infty}^\infty t_A(x, y)e^{[-\frac{i2\pi}{\lambda f(xu+yv)}]} dx dy]^2 \quad (3)$$

Since intensity is proportional to the square of amplitude, the square root of eq. 3 describes the amplitude distribution through a thin lens. A comparison of the square root of eq. 3 to eq. 2 shows that the amplitude distribution through a thin lens corresponds exactly to the Fourier transform.

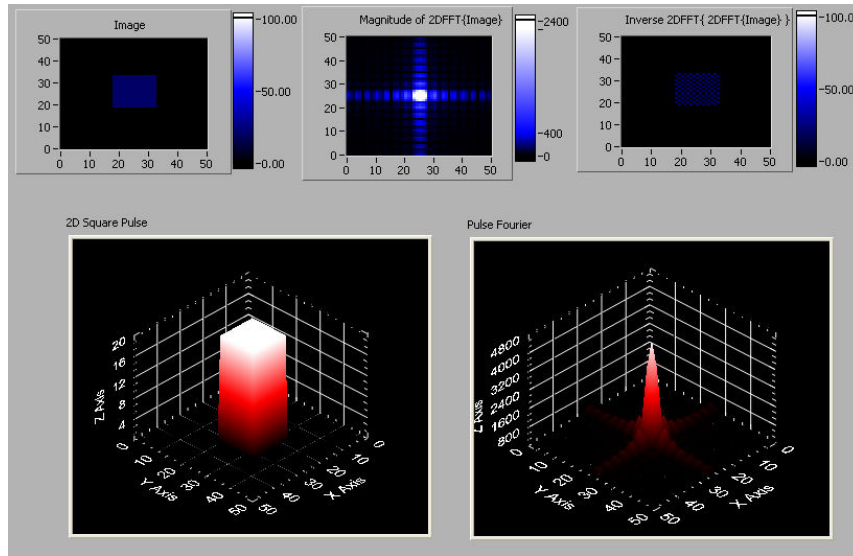


Fig. 1: Computer Simulation of the Fourier transform of a square pulse. The top row of images (left to right) shows intensity plots of the square pulse, Fourier transform, and inverse Fourier transform. The bottom row plots in three dimensions the square pulse (left) and its Fourier transform (right). Scales for all figures are in arbitrary units.

To more accurately visualize the functioning of a lens as a Fourier transformer we created a computer simulation accepting a variety of inputs and displaying their respective Fourier transforms. This simulation allowed us to view the transform of a given intensity plot and to use this information to ensure the proper operation of our lens system. Figure 1 shows the results of a computer simulation in LabView with an input of a two dimensional spatial square pulse. The computer simulation plots a given function and subsequently displays the corresponding Fourier transform. The simulation provides a surface plot, as well as the intensity plot of the Fourier transform. The computer

simulation conforms to the result found in Goodman, returning a *sinc* function as the Fourier transform of the square pulse.

1.2 Phase Modulation

In order to modify the spatial profile of the light source passing through our optical system, we had the choice of altering either the amplitude or the phase of the light wave. Phase modulation is a process which shifts the phase of an electromagnetic wave, namely light, by creating a delay in the wave through time, or in other words creating a phase shift in the light from $0-2\pi$. This is achieved practically through two basic methods, either altering the physical distance in some meaningful way (on the order of the wavelength of the wave), or altering the optical path by varying the index of refraction through which the light travels [8].

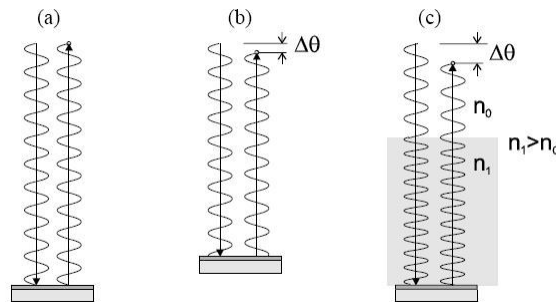


Fig. 2: Changing the physical distance traveled by a beam relative to a reference creates a phase shift $\Delta\theta$ (b). Introducing a medium with a different index of refraction into the path of the beam alters the effective optical path and creates a phase shift $\Delta\theta$ (c). Figure taken from [8].

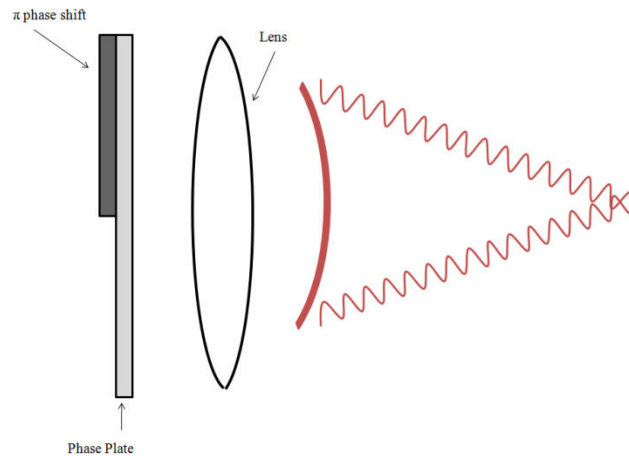


Fig. 3: Phase modulation in a laser/lens system. Both beams start out in phase. On encountering the phase plate the beam is phase shifted in the near field and made to be out of phase from the bottom beam by a half wavelength (π phase shift). The beams then pass through the lens, and add up destructively to create a dark spot at the center of the focal plane in the far field.

The phase of an electromagnetic wave is described by an imaginary exponential term $e^{i\theta}$ in the wave function, where θ is a value between $0-2\pi$. Returning to our discussion of the lens, phase modulation can be understood as attaching an $e^{i\theta}$ term to the square root of the intensity distribution through a thin lens (eq. 3). The resulting amplitude distribution, the square root of the intensity distribution, through the lens in the far field is affected by the change in phase of the laser pulse due to phase modulation in the near field. We studied a simple model for spatial modulation of a Gaussian laser beam in the near field by a phase plate. An example is depicted in fig. 3 [9].

Without phase modulation, both beams of fig. 3 would be in phase for their entire path and would add up constructively to give a bright spot at the center of the focal plane. However if the top beam is phase shifted by a half wavelength (π) then the beams would add up destructively and create a dark spot at the center of the focal plane. A simple

alteration of the phase of a laser beam in the near field can change a region of constructive interference to destructive and radically affect the amplitude distribution of the focal image in the far field.

A simple laser/lens system is perfectly suited to demonstrate the effect of phase modulation on the spatial profile of a laser beam. Our equipment accomplishes phase modulation by varying the refractive index of the medium through which the laser travels hence changing the optical path of the laser beam. Phase modulating a laser beam of Gaussian wavefronts in the near field and subsequently passing it through a lens can produce radical spatial alterations to the beam in the far field. An example of phase modulation via a SLM is shown below in fig. 4.

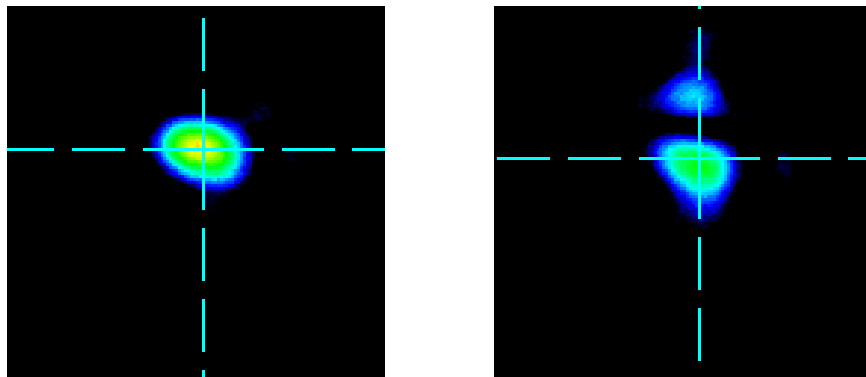


Fig. 4: Far field intensity plot of an unmodified laser beam (left) and a phase modulated beam (right)

2 Experimental

2.1 Experimental Setup

Phase modulation of a Fourier transformed laser pulse (i.e. from the near field to the far field through the use of a lens) can create some interesting pulse shapes. Our goal

was to control the phase modulation of a laser beam in order to create our own custom shapes. To accomplish this we configured a table-top optical setup as shown in fig. 5.

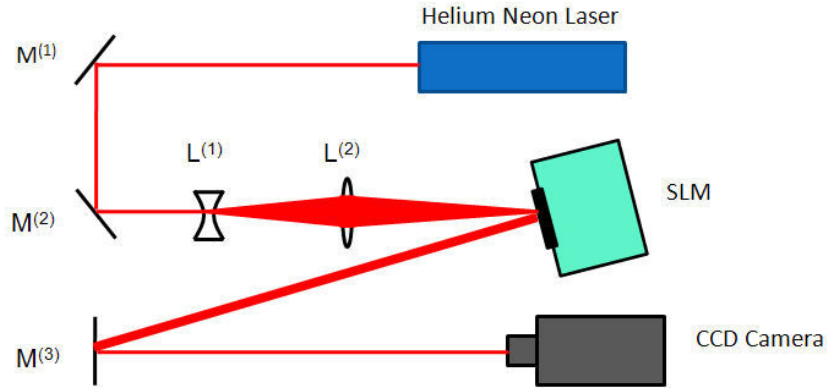


Fig. 5: Table-top Optical Setup. A Helium Neon Laser (He-Ne) is reflected off mirrors $M^{(1)}$ and $M^{(2)}$. The beam is passed through a negative lens, $L^{(1)}$, to expand it to fill the face of the SLM and then a positive lens, $L^{(2)}$, to gradually bring the beam back to a focus. The beam then reflects off the SLM and is again reflected by another mirror, $M^{(3)}$, until the beam is finally brought to a focus at the CCD camera. Distances are not drawn to scale.

Our table-top setup uses as the source a helium neon (He-Ne) laser. While this laser is not the femtosecond laser that is the focus of our experiment, a He-Ne laser suffices as a good approximation and affords us more experimental ease due to its relatively weak power output of 10mW versus the 500mW mean power output of the femtosecond laser. To keep our setup compact, we use a system of mirrors to direct the laser beam. First, the laser is sent through a negative lens, thus expanding it to fill the SLM. Then it travels through a positive lens to gradually bring it back to a focus. Next, the altered light is sent to the Spatial Light Modulator (SLM) and reflected to the final mirror. Finally, the reflected light from the final mirror terminates at the Charged Coupled Device (CCD) camera. We set up our system so that this focus occurs at the face of the CCD camera to give us a nice far-field beam shape on the camera. Furthermore,

the focus of the laser passes through several neutral density filters before it reaches the camera lens which act to decrease the intensity of the light incident on the CCD camera.

2.2 Spatial Light Modulator

The SLM is our tool for phase modulation, a device which controls and alters light on a pixel-by-pixel basis. On the face of the SLM is an array of nematic liquid crystals, each with the ability to pass a discrete voltage through them. These voltages deform the nematic liquid crystal by rotating the molecules which compose the liquid crystal. The deformation of the constituent molecules affects the birefringence of the liquid crystal. Birefringent materials like liquid crystals possess different refractive indices dependent on the polarization of the incident light. As a voltage is placed on the nematic liquid crystal, its molecules rotate in response to the electric field, thereby changing the refractive index of the incident light. Different refractive indices correspond to different phase shifts; altering the voltage placed on the liquid crystal allows us to modify the effective phase of the laser beam and in turn modify the beam's spatial profile.

Our SLM contains 262,144 (512x512) discrete elements. Each one is programmable to any of 256 discrete voltage stages. The SLM functions by reading in a grayscale (pixel values of 0-255) phase profile from the computer and translating the pixel values of the phase profile to voltages in the liquid crystals of the SLM. This proceeds through the use of computer software run in a LabView environment. The SLM does not respond to the polarity of the electric field put on the liquid crystals, so pixel

values of n and $255-n$ result in the same phase shift. This limits the relevant pixel values to 0-127.

Since each SLM is produced individually there are slight production differences which can cause the SLM to react differently than expected for a given pixel value in the phase image. Possible defects of production require a calibration procedure to ensure that the SLM's response to a pixel value in the phase image conforms to what we would expect. An interesting way to test the calibration of the SLM involves using the SLM as one of the mirrors in a Michelson Interferometer [10]. A schematic view of the SLM-Michelson setup is shown in fig. 6.

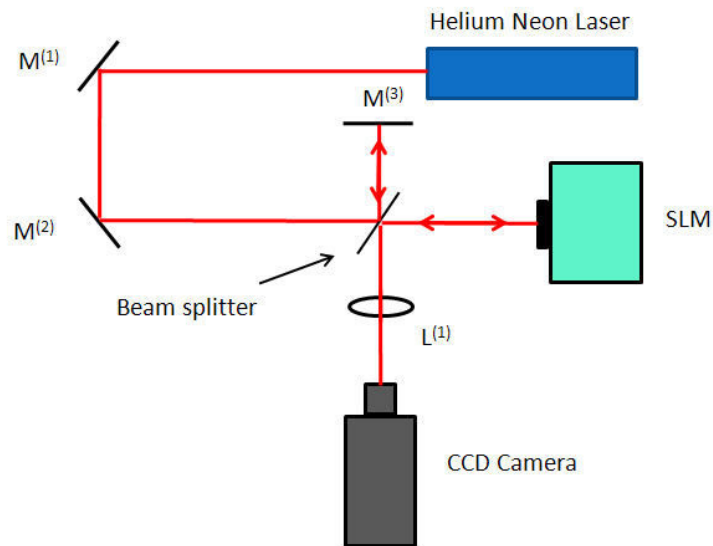


Fig. 6: Michelson Interferometer with SLM. The He-Ne beam reflected off $M^{(2)}$ passes through a non-polarized beam splitter. Instead of a mirror at the end of both arms of the interferometer the SLM replaces the mirror in one of the arms. The beams combine and are passed through a lens which brings the beam to a focus at the CCD camera.

With the SLM situated in the Michelson setup we send phase images consisting of black (pixel value of 0) to the top portion and varying shades of solid gray (pixel values 1-127) to the bottom portion to the SLM and compare the effect on the interference fringes in the respective halves, as shown in fig. 7b.

In fig. 7b are shown the familiar interference fringe patterns that arise from a Michelson Interferometer. We can measure qualitatively the degree of the phase shift by comparing the offset of the top portion of the fringe pattern to the bottom. A visual inspection reveals that the maximum (π) phase shift occurs approximately at grayscale value 54, with the corresponding interference fringe pattern shown in fig 7c. Additionally, our results confirm that pixel values 0 and 127 represent an entire “cycle” of phase shifts ($0-2\pi$) which conforms to the published result [11].



Fig. 7a: Phase Image 0/ n , with the 0/ n notation representing a pixel value of 0 for the top half of the phase image and a pixel value of n for the bottom half

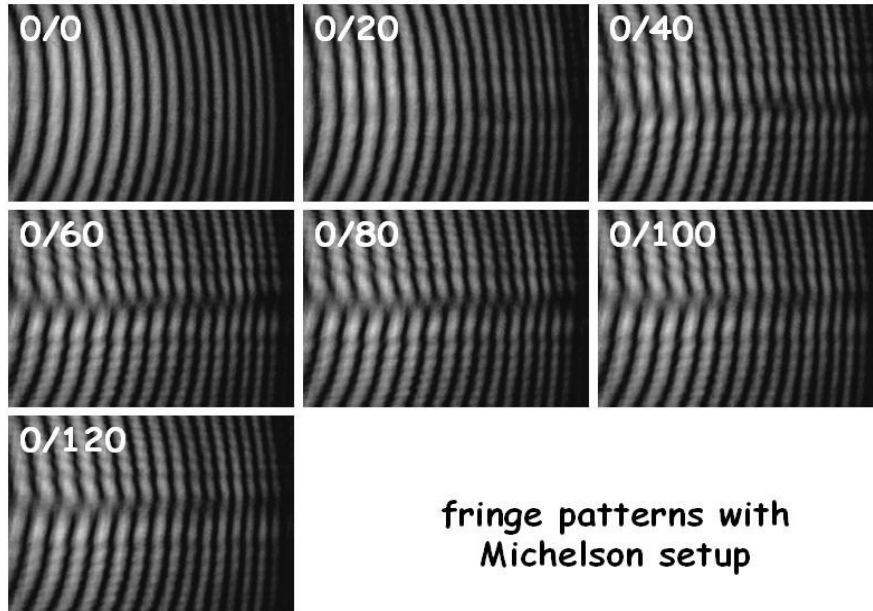


Fig. 7b: Interference Fringes of various phase patterns. As the grayscale pixel value is increased from 0-127 the fringes “walk” through the cycle of phase shifts until they return to perfect alignment.

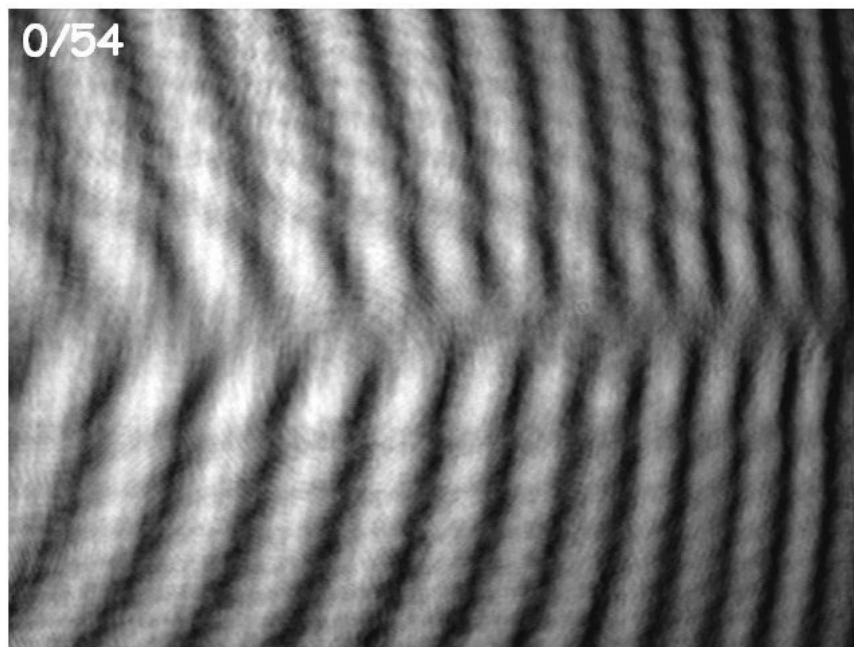


Fig. 7c: Close up view of the interference fringe pattern caused by the phase image 0/54. The fringe pattern of this phase image demonstrated the maximum (π) phase shift as determined by visual inspection.

3 Genetic Algorithm

A genetic algorithm is a “survival of the fittest” programming technique which derives its name from the biological process that it mimics. Members of a population compete for existence, and those members who are better suited survive and proliferate. Instead of calculating a solution based on set formulas and initial conditions, genetic algorithms utilize feedback to iteratively solve a problem. These types of algorithms are commonly used in physics, biology, economics, and in the field of cryptography. The optics community relies heavily on genetic techniques. For example, experiments have been conducted utilizing genetic algorithms to maximize third-harmonic light generation [12].

3.1 Theory

The basic building blocks of a genetic algorithm are the population and the fitness criteria [13]. The population is a list of all acceptable solutions to the given problem. The fitness criteria determine how close a solution is to the target value. The algorithm will test these possible solutions against the fitness criteria, keeping a solution only if it is an improvement to the previous best guess. This loop continues until the fitness criterion has been satisfied and the algorithm converges on the most acceptable solution.

There are a few variants of genetic algorithms. The most frequently used types include “elitist” type algorithms and “population” type algorithms. Elitist algorithms save the best solution from all the iterations, searching for an improved solution only from later iteration. Population algorithms generate an initial population of solutions, test them for their respective fitness, and then breed the most successful members of the

population together to create a new population of solutions. These types of algorithms exchange information from parent solutions to breed a new solution through a process known as crossover [13]. Both variants have drawbacks. Since elitist algorithms only make improvements from subsequent guesses, they could become stuck on a local maximum, and not reach the preferred global maximum, a process known as “hill-climbing” [13]. Population algorithms eliminate this problem since new solutions are randomly bred. However, depending on the nature of the solution the amount of calculations involved in producing a population can be massive, thus efficiency becomes a concern.

3.2 Laser Application

We employed a genetic algorithm technique to modify the spatial profile of a laser beam. The components of our system were the same as the optical setup of fig. 5 (namely a laser, a series of optics to refine the image, a computer controlled SLM, and a computer controlled CCD camera). Through the use of a computer we altered the phase written to the SLM and observed the alterations to the beam through images taken by the CCD camera.

The first requirement of the algorithm design was to be able to control the main constituents of the optical setup, the SLM and the CCD camera. We chose to control both pieces of equipment with LabView programming software. Our SLM was packaged with a software development kit which allows the user to write a program that sends phase images stored on a sequence list to the SLM. The sequence list holds the file location information about each phase image and no image can be sent to the SLM unless

it is included in the sequence list. The first programming challenge required us to manipulate the software development kit to force the SLM to enter a “continuous download” mode. In continuous download mode the SLM can write custom phase images from the computer to the liquid crystals at a max rate of 1 kHz and bypass the sequence list. We modified the LabView code to run in continuous download mode at this maximum speed of 1 kHz.

The computer reads the CCD camera through a standard framegrabber program written in LabView. Since the genetic algorithm will perform calculations with these captured images, the images need to be as small and as simple as possible to make the algorithm run efficiently. The raw images captured by the CCD camera are too large and contain too much background interference to be useful for calculations. To make the captured images simpler we customized the framegrabber code to allow us to specify the size of the image taken in, and further modified the code to set a threshold pixel value. If a pixel in the captured image is greater than or equal to the threshold value then this pixel is set to a solid blue color (R=0, G=0, B=255 in RGB format), if it is less than the threshold value, the pixel is set to black (R=0, G=0, B=0 in RGB format). An example of a modified camera image is shown in fig. 8. Our image processing code takes images from the CCD camera and returns them reduced and simplified.

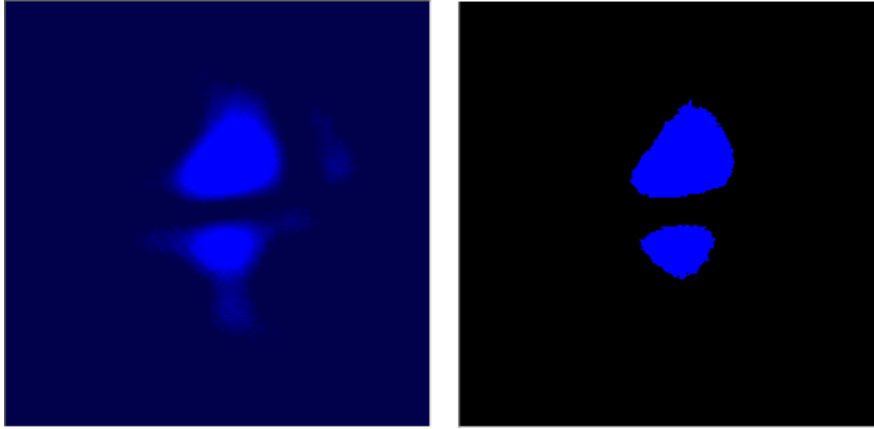


Fig. 8: Normal (left) and processed (right) CCD camera image of a modified laser beam. The processed image contains only two colors, black and blue, and has the background interference removed leaving only the two main beam spots.

With successful control of the SLM and the CCD camera achieved, we then turned to the design of the genetic algorithm. The population used by our algorithm is simply the phase images to be sent to the SLM. As determined during the calibration of the SLM, the pixel values in potential phase images need only be grayscale values between 0-127. Using this information reduces the processing time of the computer, since the available pixels are reduced by a factor of two. The size of the phase images can also be altered. The SLM allows a maximum phase image of 512 x 512 pixels, or a one-to-one mapping of phase image pixel to liquid crystal channel. The software that writes the phase image to the SLM contains a built in scaling algorithm, so phase images smaller than 512 x 512 can be up-scaled and then passed to the SLM. This scaling algorithm works best if the images are even factors of 512, so square phase images of 32, 64, 128, or 256 pixels are appropriate. In generating the population of solutions for the genetic algorithm, an image of size 256 x 256 contains 4 times fewer pixels than its 512 x 512 counterpart. Taking advantage of the scaling algorithm integrated in the SLM software further reduces computational time.

We determine the fitness criteria for our algorithm by comparing camera images to a predefined target image. To test the fitness of a phase image we need to see its result on the shape of the laser pulse. A comparison of the current image to the pre-defined target image gives us our measure of fitness. In this comparison we compute the absolute value of the difference between the two images and sum these differences over all of the pixels. If this number is smaller than the previous value the current iteration has gotten closer to the goal and is deemed more “fit.”

After defining the population and fitness criteria we needed to choose which type of genetic algorithm to employ. The efficiency of the algorithm is of the utmost importance to its successful implementation in the laboratory. A population algorithm requires the creation of an initial population of solutions, a test to compare these solutions, and breeding of the fittest solutions, all of which are computer intensive. For example, if we used a population of 100 randomly generated phase images each 128 x 128 pixels in size, the computer would need to generate, compare, and breed 1,638,400 different pixels during each iteration of the genetic algorithm. This number ignores the processing demands of the captured images, which would more than double the total number of pixels that need to be processed.

Conversely, each iteration of an elitist algorithm requires only one phase image to be produced, sent to the SLM, and fitness tested, and thus requires far fewer computations per iteration. The benefits to efficiency garnered by the elitist algorithm made it the preferable option. A schematic view of our elitist algorithm is represented in fig. 9. Our algorithm follows an intuitively simple process. First, a phase image is generated and sent to the SLM. The resulting spatial modification is then captured by the

CCD camera, and tested for fitness by the computer. If the modification is an improved solution, then the phase image is saved and another section is mutated. If not, then the previous best phase image is mutated again. This loop continues indefinitely until the program reaches perfect fitness or is terminated by the user.

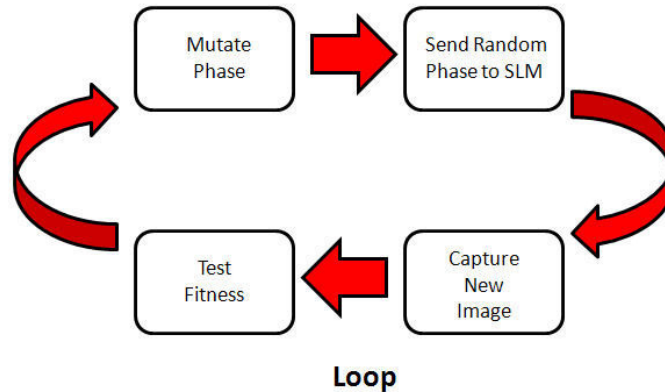


Fig. 9: Schematic view of our Elitist Algorithm. A random phase image is initially generated and sent to the SLM. The resulting spatial modification to the laser beam is captured by the CCD camera and tested for fitness. Next the phase image is mutated according to the fitness score and sent to the SLM. This process repeats in a loop until the algorithm reaches perfect fitness or is terminated by the user.

While the genetic based loop structure of our elitist algorithm is constant, our algorithm is written in a generic fashion. We leave ourselves the freedom to define the size of the phase image being sent to the SLM and allow for several different methods to mutate the phase images throughout the algorithm iterations. Adjustment of these two important algorithm parameters will allow us to adjust the convergence time and efficiency of our algorithm. Investigation of the effects of these parameters on algorithm efficiency is vital.

4 Results

To apply our algorithm to the optical setup of fig. 5 we proceeded in a two-step fashion. First, we pre-loaded a target image and tested the algorithm. Second, after obtaining an optimized image through this method we moved to a user-generated target image and re-tested the algorithm. Through both steps, we tested variations in the size of the phase image and the method of mutation in order to improve the efficiency of the algorithm.

4.1 Pre-Load Generic



Fig. 10: A 0/64 Phase Image (left) is written to the SLM and results in a two-spot focal image (right) captured by the CCD camera.

The initial trials of the algorithm involved using a pre-defined target image. We chose a two-spot focal pattern, which is generated by a zero phase shift on the top half and a π phase shift on the bottom half. From the calibration results we knew that this type of spatial pattern comes by writing a half gray, half black phase image to the SLM.

We captured this image using a variation of the framegrabber software used in the design of the elitist genetic algorithm.

After acquiring the target image, we ran the algorithm in the most generic fashion that our system would allow. Generation of the initial phase guess is done completely randomly with each pixel constrained to a value between 0-127. The code which mutates the phase images is set to choose a random pixel each iteration and to change the selected pixel to a random value between 0-127. We ran several trials of this form, each time adjusting the size of the phase image being sent to the SLM. Figure 11 shows the results from two initial trials using the completely generic algorithm.

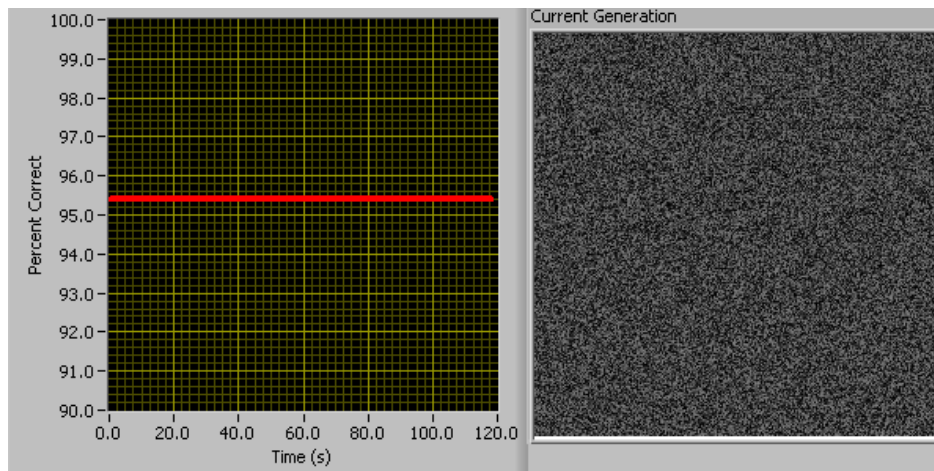


Fig 11a: Plot of percent correct of the best CCD image versus time (left) and the current 256 x 256 pixel phase guess (right).

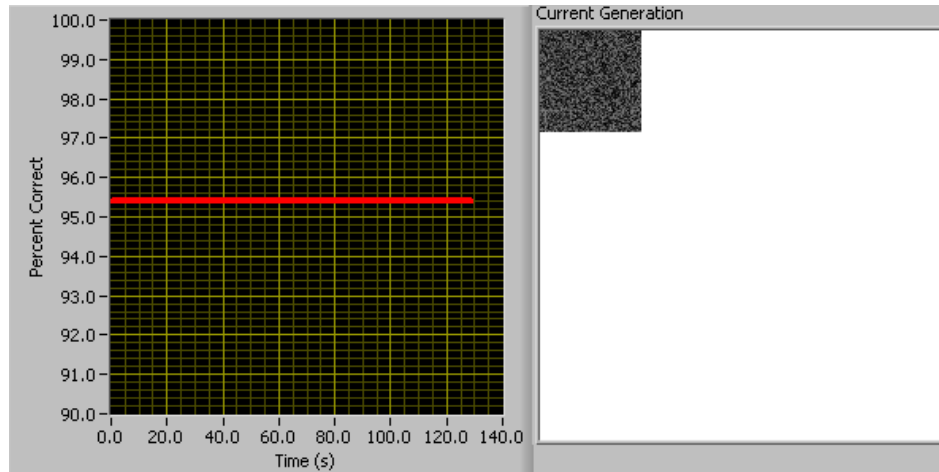


Fig. 11b: Plot of percent correct of the best CCD image versus time (left) and the current 64 x 64 pixel phase guess (right).

For phase images greater than 8 x 8 pixels, mutating one pixel per iteration has no practical effect on the spatial modification, because such minor alterations do not show up as significant modifications of the laser images captured by the CCD camera. Since these differences cannot be detected, our algorithm is unable to determine if a given phase mutation was an improvement over the previous guess, and the algorithm continues to run without ever converging to a solution.

As a result, we tried mutating many pixels per iteration. These trials were done in exactly the same fashion as before with the sole exception that during each iteration multiple pixels were randomly mutated. The results were again discouraging. If we set the number of pixels changed each iteration (i.e. mutation rate) to a small value like 100, the algorithm is still unable to make a significant change to the phase image. Since the mutation algorithm randomly chooses which 100 pixels to mutate, little visible change is produced in the phase image each iteration. These random, single mutations cause no significant change to the spatial profile of the laser beam in aggregate, and are unable to be detected by the CCD camera. If we set the mutation rate to a much higher number, we

increase the practical “effect” of each mutation on the spatial profile. Unfortunately, the randomness of the mutation algorithm requires too much processing by the computer and the algorithm loop takes nearly 30 seconds per iteration, too slow for practical application.

4.2 Pre-Load Block

For there to be any hope of modifying the laser image, we needed to mutate a substantial number of pixels per iteration. The challenge was to do this in such a way as to minimize the amount of calculations per iteration. The most obvious solution was to mutate a “block” of pixels. Instead of changing 100 pixels of random location, we mutated a 10 x 10 block of pixels. This approach remedies both of the previous problems. The block approach mutates enough pixels to cause a noticeable change in the laser image, and the processing demands of creating a block of a solid grayscale value are much less intensive than mutating the equivalent number of random pixels.

We altered the genetic algorithm so that mutations to the phase image would be accomplished through a block pattern. The new algorithm mutates the phase images by creating a block of a random grayscale value between 0-127 and inserting it into the phase image. We designed the mutation method so that blocks of a random pixel value are inserted in an orderly fashion from top to bottom and left to right. This cycle continues until the bottom right block is placed, where the cycle is started over. An example of this type of mutation is shown in fig. 12.

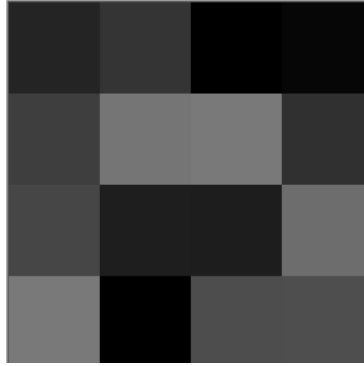


Fig. 12: Block Mutated Phase Image (256 x 256). The grayscale values range from 0-127 and the blocks are 64 x 64 pixels.

After implementing the block mutation method into our genetic algorithm we ran several trials, each time varying the size of the phase image sent to the SLM. Now that the processing limitations of the previous trials had been removed, we discovered that the size of the phase image sent to the SLM does not significantly affect the processing speed. At first this result was surprising since larger phase images require more calculations due to the larger number of pixels. However, the speed of the algorithm is constrained by the refresh time of the SLM (1 kHz) and the time needed to compare the CCD image against the target for fitness. The block mutation method operates so efficiently that any size of phase image can be used without a noticeable effect on algorithm speed since mutations are done faster than they can be written to the SLM. We chose a phase image of 256 square pixels for all further trials.

Now that we had determined the size of the phase image, we tested and compared results from trials with varying block dimensions. Similar to the phase image size itself, the size of the blocks can only be even factors of the phase image size so that the blocks

fit evenly on the phase image. We compared the convergence time of the algorithm using blocks of 8×8 , 64×64 , and 128×128 pixels, with the results shown in fig. 13.

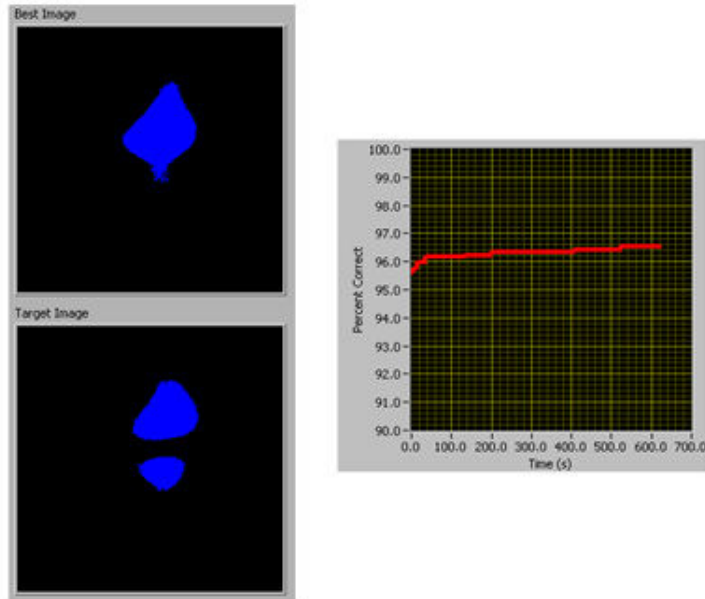


Fig. 13a: Resulting image after 10 minutes (top left) against the target image (bottom left) using block mutation of block size 8×8 pixels. The plot (right) shows the percent correct of the best CCD image versus time.

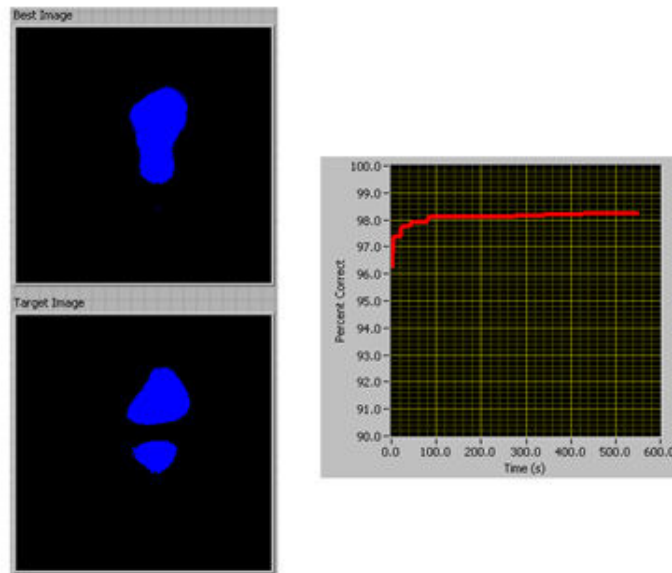


Fig. 13b: Resulting image after 10 minutes (top left) against the target image (bottom left) using block mutation of block size 64×64 pixels. The plot (right) shows the percent correct of the best CCD image versus time.

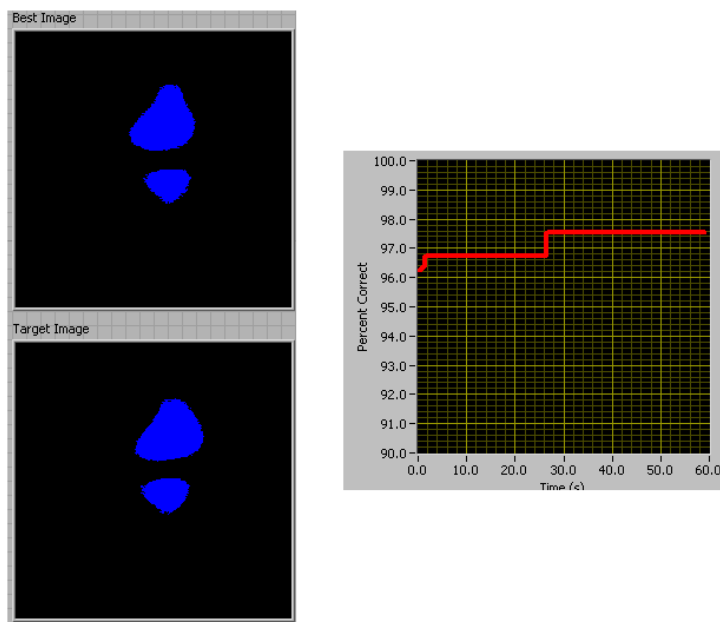


Fig. 13c: Resulting image after 1 minute (top left) against the target image (bottom left) using block mutation of block size 128 x 128 pixels. The plot (right) shows the percent correct of the best CCD image versus time.

Using an 8 x 8 block size we saw little improvement to the image after running the algorithm for 10 minutes. This is not surprising since the 8 x 8 blocks contain 64 total pixels, fewer total pixels than even the 100 pixels we mutated in previous trials. Mutating such a small number of pixels each iteration creates a negligible change to the laser beam and hence inhibits the algorithm from noticing any fitness differences in mutated phase images. Block sizes of 64 x 64 and 128 x 128 pixels produced more promising results, with the 64 x 64 pixel block trial reaching 98% fitness in about two minutes, and the 128 x 128 pixel block trial reaching 98% in less than one minute. Block sizes of 16 and 32 produced working algorithms but their convergence times were much slower than their 64 and 128 block counterparts and were not useful.

4.3 Draw Block

From algorithm trials with a pre-defined target image we concluded that to maximize algorithm efficiency the size of the phase image should be 256 x 256 pixels and that the phase image mutation method should be blocks with a size of 64 x 64 or 128 x 128 pixels. Given these results, we turned to the second stage of our testing which was to implement a user-generated target image into our algorithm.

The simplest way to implement a user-generated target image was to create a drawing program which allows the user to draw a target image on a computer through use of a mouse. Our drawing program prompts the user to draw their target image before starting the algorithm. The image is then saved and used as the target image with which to compare CCD images during that run of the genetic algorithm.

We tested our new algorithm with block mutations of 64 x 64 and 128 x 128 pixels. In both tests the drawn target image was two parallel spots separated vertically. Both choices of block size produced working algorithms which converged in a short amount of time. As shown in fig. 14, the 128 block test converged to 95.3% fitness in about 12 seconds and the 64 block test converged to 98.8% fitness in about 35 seconds. Although both algorithms converged rather quickly with high levels of fitness it is clear from an examination of fig. 14 that the 64 block algorithm produced a best image that is visually much closer to the target image. This result is expected since the 128 block algorithm only mutates the phase image through four blocks. The phase of the laser beam can only be altered by quadrant since the blocks are so large relative to the size of the phase image. The less control the algorithm has over the phase images, the less

precise images it will be able to produce. Therefore, a mutation block size of 64 maximizes both the efficiency of the algorithm and the precision of the modified laser images it can produce.

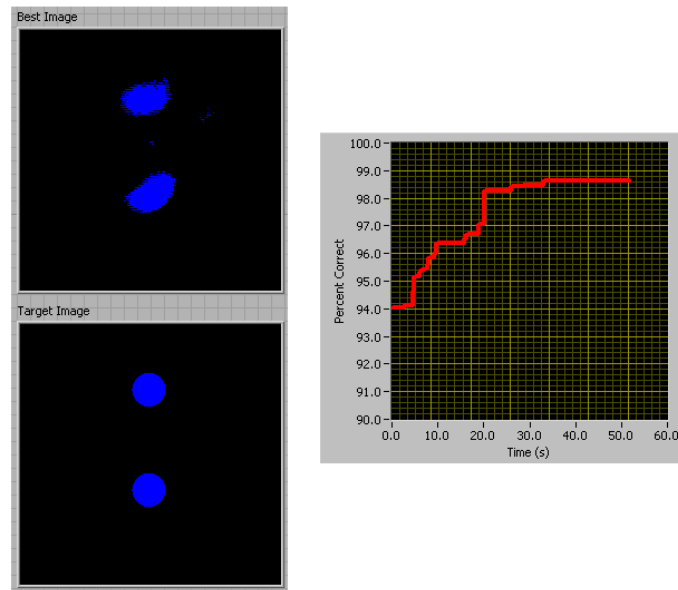


Fig. 14a: Resulting image after 1 minute (top left) against the drawn target image (bottom left) using block mutation of block size 64 x 64 pixels. The plot (right) shows the percent correct of the best CCD image versus time.

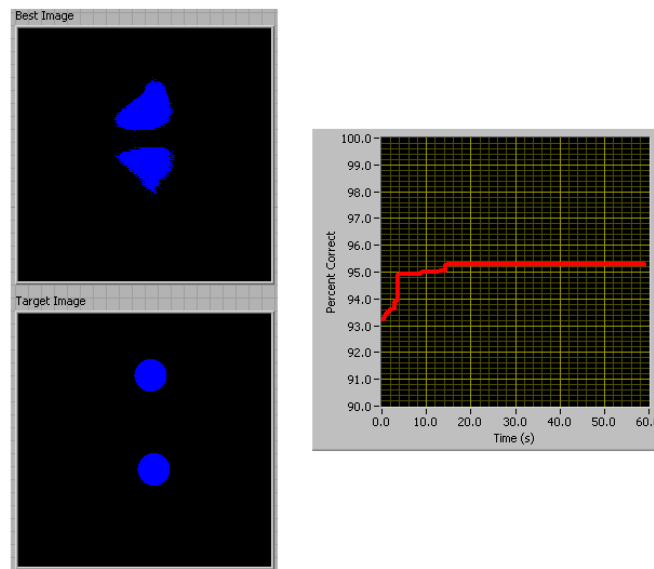


Fig. 14b: Resulting image after 1 minute (top left) against the drawn target image (bottom left) using block mutation of block size 128 x 128 pixels. The plot (right) shows the percent correct of the best CCD image versus time.

5 Conclusion

We have designed a genetic algorithm which through phase modulation of a laser beam can create custom laser images. The central components of our setup are the laser source, the SLM, and the CCD camera. An algorithm run in a LabView environment controls the SLM and the CCD camera. The algorithm dynamically mutates the phase image written to the SLM and captures the corresponding alterations to the laser image with the CCD camera. The captured images are then compared to a pre-defined target value and given a fitness score. The algorithm runs in a loop continuously searching for solutions of better fitness.

Our algorithm is written in a generic form and there are several important parameters which can be altered. We tested various permutations of these parameters with the goal of maximizing algorithm speed and convergence time and several important results emerged. First, an elitist algorithm is preferable to a population algorithm. The processing time required to create a sufficient population of solutions renders a population algorithm impractical in a laboratory setting. Second, a block mutation method is preferred to a random mutation method. The net change of one pixel in the phase image is negligible which prevents the CCD camera in our setup from detecting changes in the camera image. Consequently, larger pixel alterations must be performed each iteration and modifying a block of pixels is the fastest way to accomplish this.

The size of the phase images and the mutation blocks also affect the success of the algorithm. Since the phase image creation and mutation methods run faster than the other components of the algorithm, the size of the phase image is only constrained by the

limitations of the SLM. We chose a phase image of 256 x 256 pixels to maximize visual clarity in the algorithm interface. The mutation blocks must be large enough to cause a noticeable change in the laser image, but small enough so not to limit the potential image precision. A block size of 64 x 64 pixels maximizes both algorithm speed and potential image precision.

There are many exciting non-linear optics experiments which are particularly suited for the combination of an SLM and a genetic algorithm. Many of these experiments pertain to the concept of continuum generation. Continuum generation is the result of light passing through an optical medium, like an optical fiber or a crystal, and inducing self-phase modulation. Self-phase modulation is the change in the refractive index of a medium as a result of the incoming light source. The varying refractive index creates shifts in the incoming pulse's phase which alter the frequency spectrum of the pulse and produce the continuum output.

Other types of experiments study second harmonic generation, a process where photons with a given frequency interact in a non-linear material and combine to generate new photons of double the initial frequency. High intensity light sources are needed for this interaction to occur, and laser beams would provide sufficient intensity. A picture of second harmonic generation through a non-linear crystal is shown in fig. 15.

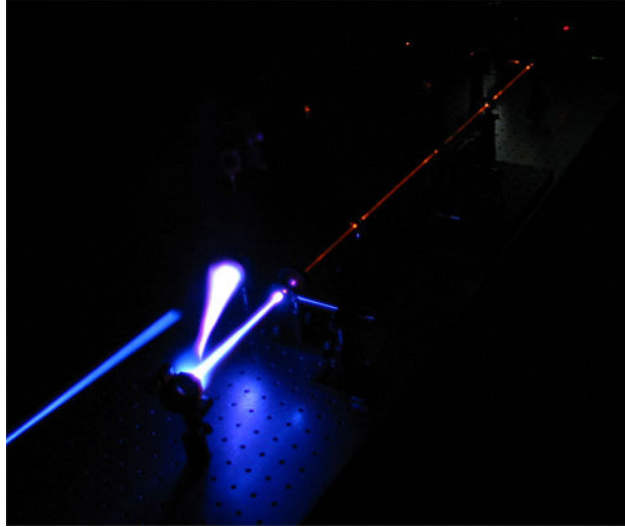


Fig. 15: Second harmonic generation through a non-linear crystal.

Experiments have been reported in the literature which demonstrate continuum generation and second harmonic generation [14,15]. We would build on this work and use our genetic algorithm to control the output in the far field by modifying the spatial profile of the laser beam in the near field. For continuum generation, a simple experiment would involve using the genetic algorithm to maximize the spectral bandwidth of the continuum by varying the phase of the incident laser pulse. Second harmonic generation would follow a similar approach, with the exception that we would maximize the intensity of the second harmonic generation rather than the spectral bandwidth of the output.

References

1. T. Pfeifer, R. Kemmer, R. Spitzenpfeil, D. Walter, C. Winterfeldt, G. Gerber, and C. Spielmann, "Spatial control of high-harmonic generation in hollow fibers," *Opt. Lett.* **30**, 1497-1499 (2005)
2. S. Postma, P. van der Walle, H. L. Offerhaus, N. F. van Hulst, "Compact high-resolution spectral phase shaper," *Rev. of Scientific Instruments* **76**, 123105 (2005)
3. F. K. Fatemi and M. Bashkansky, "Generation of hollow beams by using a binary spatial light modulator," *Opt. Lett.* **31**, 864-866 (2006)
4. C. Trallero-Herrero, J.L. Cohen, T. Weinacht, "Strong-Field Atomic Phase Matching," *Phys. Rev. Lett.* **96**, 063603 (2006)
5. D. Oron, Y. Silberberg, "Spatiotemporal coherent control using shaped, temporally focused pulses," *Opt. Express* **13**, 9903-9908 (2005)
6. E. Hecht, *Optics 4th edition*, (Addison Wesley, San Francisco, 2002).
7. J. Goodman, *Introduction to Fourier Optics 3rd edition*, (Roberts & Company, Englewood, 2005).
8. E. Hällstig, Nematic Liquid Crystal Spatial Light Modulators for Laser Beam Steering. Diss. Uppsala Univ., 2004. 01 Oct. 2006
<<http://publications.uu.se/theses/abstract.xsql?dbid=4693>>.
9. L. Casperson, "Spatial Modulation of Gaussian laser beams," *Optical and Quantum Electronics* **10**, 483-493 (1978)
10. J. Harriman, A. Linnenberger, S. Serati, "Improving spatial light modulator performance through phase compensation," Unpublished (2005)
11. Boulder Nonlinear Systems, Inc., "XY Phase and PhaseFlat Series SLM User Manual," Rev. 3.1 (2005)
12. T. A. Planchon, W. Amir, J. J. Field, C. G. Durfee, J. A. Squier, P. Rousseau, O. Albert, G. Mourou, "Adaptive correction of a tightly focused, high-intensity laser beam by use of a third-harmonic signal generated at an interface," *Opt. Lett.* **31**, 2214-2216 (2006)
13. Mitchell, Melanie. An Introduction to Genetic Algorithms. Cambridge: MIT P, 1996. 166-174.
14. B. von Vacano, W. Wohlleben, M. Motzkus, "Actively shaped supercontinuum from a photonic crystal fiber for nonlinear coherent microspectroscopy," *Opt. Lett.* **31**, 413-415 (2006)
15. T. Pfeifer, R. Kemmer, R. Spitzenpfeil, D. Walter, C. Winterfeldt, G. Gerber, and C. Spielmann, "Spatial control of high-harmonic generation in hollow fibers," *Opt. Lett.* **30**, 1497-1499 (2005)