Miralens

Matin Raayai Ardakani
Quan Do
Josh English
Luke Horgan
Sarada Symonds

Faculty Advisor: Prof. John Kimani

April 2019

Abstract

For decades, holograms have been a staple of science fiction. In reality, thus far, an affordable 3D solution for interacting with digital media has yet to be discovered. Our team has created a cost-effective holographic system called Miralens: A high frame rate full-color projector projecting onto an intangible volumetric display. Miralens manifests 3D digital images rendered by a host computer in physical space for its users to see and interact without any help from additional apparatuses like glasses or gloves. In other words, with their naked eyes, users can walk around the system to observe a seemingly real 3D object. Touching the object, however, will reveal nothing but a splash of light.

To create a holographic illusion for the human eyes, we invented a 1200Hz projector made from a 1080p 240Hz LCD segmented into 5 grids each controlled by a custom backlight. By redirecting and converging the generated frames in each LCD grid into a single frame using custom prisms, it's possible to project hundreds of digital images onto a rotating dynamic volume spinning at around 15Hz, tricking the human eyes into seeing a 3D object. Finally, for the users to be able to touch and interact with the volumetric image, a Mirascope is used to create an optical illusion of the dynamic volume in thin air. Practical applications include dynamic mapping in performance art, interactive advertisement, and intuitive display for productivity. More importantly, we want to inspire the future generation, to let it be known that it's possible to bring science fiction to reality and to redefine what is possible.

Contents

1 Introduction

1.1 Problem Formulation

Courtesy of popular science fiction franchises such as *Star Trek*, *Star Wars*, and *Iron Man*, holograms have captured the public's imagination for decades and remain an area of active research. This project was started with the goal of creating a holographic 3D display in order to render digitally processed images. While the current user experience relies heavily on flat electronic displays, the physical world is made of three dimensional objects. The ability to render a realistic volumetric image allows for improved depth perception and enables multiple users to view the image simultaneously from various angles.

1.2 Existing Products and Patents

There have been several attempts to reproduce the types of holographic images seen in popular fiction, yet none have become widely adopted and research is still ongoing. One popular approach to 3D volumetric displays is to project light onto particles normally unseen by the human eye, and there are several patents registered with the United States Patent Office related to this method. For instance, patent US20120274907A1 describes a 3D volumetric display which projects light onto projected electromagnetic particles.

Another patented 3D volumetric display is the *Illumyn 3-D Display*, developed by researchers at University of Rochester, which projects an image onto cesium vapor waves using lasers [6]. However, this approach requires a container full of particles on which to project the light, meaning the effect is not truly holographic. Furthermore, researchers using this method were not able to produce larger, high quality images since they were relying on lasers to color each particle.

Other research has focused on using high speed projectors to improve the image quality, which is the approach we've decided to pursue. Since projectors are not as precise as lasers, these types of volumetric displays use dynamic surfaces in order to create a 3D effect. Currently, off-the-shelf projectors do not run at high enough speeds to create the 3D vision effect, including projectors released as addons for hacker boards.

Researchers at the University of Southern California were able to reprogram a DLP projector in order to get a higher frame rate and create a 3D image using a rotating surface, but could only project in one color [11]. The University of Tokyo was able to develop a high speed, greyscale projector for use in dynamic mapping, but they have not yet applied this technology to 3D volumetric displays [13]. Our contribution to this research will be developing a 3D volumetric display that uses RGB to create high resolution images.

1.3 Solution

Our solution is to create a volumetric full color 3D display that renders digital images in physical space for the users to view without any additional apparatus like glasses or headsets. We completed 3 milestones. First, we assembled a full-color monitor with custom backlight, refreshing at a rate fast enough to display multiple cross sections of the 3D image simultaneously. Second, we designed custom optics to redirect and overlay projected frames from such display, effectively inventing a high frame rate projector. Third, we focused the frames from the high-speed projector onto an oscillating screen to manifest a 3D image, and used a mirascope to display that image in thin air, resulting in a hologram.

1.4 Impact

As this project contains three distinct deliverables, we would be remiss if we did not address the impact of each of these parts separately. First, let us consider potential applications for a high frequency full spectrum projector. Obviously, this technology could be used in place of any high frequency projector which lacks color. This would enable devices which use this type of projector to accomplish their objectives and take advantage of color. While certain applications require only black and white projections, many other applications would benefit from color. Specifically, applications where human interaction is required, or where humans must distinguish between points in a projection would benefit from the use of color.

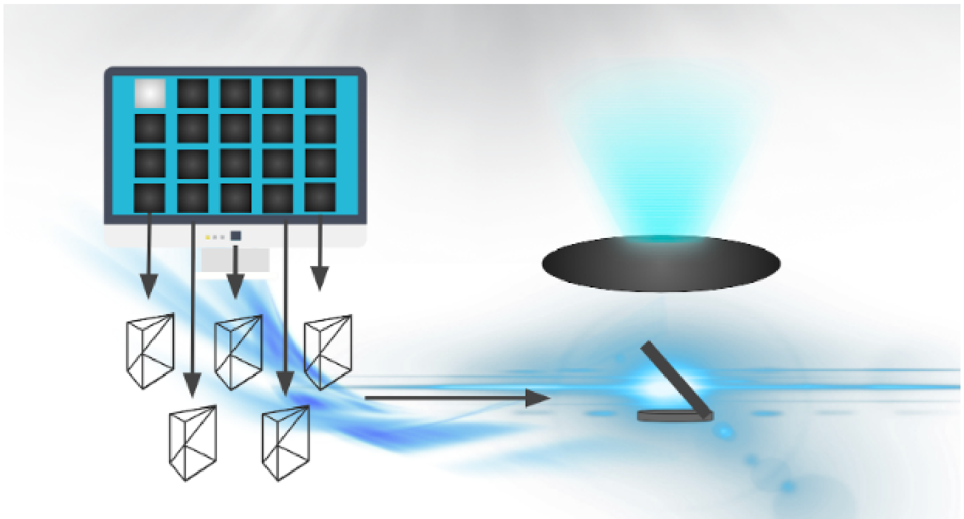


Figure 1: Assembly overview

Additional uses for a high frequency full spectrum projector includes usage for high variability dynamic projection mapping. Current applications for dynamic projection mapping are used across industries, and can even be found in performance art. There are limitations to the current technology however. Because full spectrum projectors are commercially available at 120 Hz, in situations where the projection must be mapped to a fast moving or highly-variable target, these projectors fail. For example, our projectors could be used to map images to high-velocity athletes or performers. While our high frequency projector can map to these targets, the resolution of the projected image could be compromised. This is a potential area for future work.

Second, let us consider the potential uses of the volumetric display. Although this does not fit the technical definition of a hologram, it is essentially a hologram. The applications of a hologram are potentially wide reaching, but we think the most convincing applications for this technology exist for pre-imaging and a virtual reality/augmented reality replacement. A doctor or hospital could use this technology to practice surgical procedures before entering the operating room. Additionally, this technology could be scaled up and improved upon to produce a holographic room. When inside this room, entire environments could be simulated with this hologram technology, giving an unprecedented level of immersion in the space of digital computing.

Finally, let us discuss the impact of integrating this hologram technology with user interaction. Again, the most convincing use of this technology is in the field of pre-imaging and hologram reality. Instead of merely seeing a holographic image, users would be able to interact with the digital holographic world. This would allow user interaction with the digital world in a 3D environment, rather than a 2D one. Instead of designing parts with the use of CAD software, engineers could enter a design studio and interact with a holographic rendering of their design. Additionally, students or designers could test the interaction and viability of their designs in real time.

2 Analysis and Design

2.1 Princple of Operation

A CRT monitor [7] works by scanning, with great rapidity, a bright electron beam over a screen. By convention, these scan lines are drawn from left to right across the screen, and then from top to bottom. At no point is the entire screen illuminated, nor even a single scan line bright across its entire width. The electron beam illuminates only points on the screen, and it is our persistence of vision [7] which is responsible for rendering in our minds a cogent image [9]. That is, where only points of light flash before us, we perceive a picture as though it were painted on canvas. If we could process visual input at 30 thousand frames per second, this trick would not work, and we would see only a progression of colored dots.

The working principle of a volumetric display is the same, but instead of flashing 1-dimensional pixels in lines to trace a 2D image, we instead flash a series of 2-dimensional cross sections to trace a 3D image, and we rely again on our persistence of vision [7] stitch these cross sections together. Our design may be considered in two physical parts, each of which is itself a substantial technological challenge. First, we require a projector capable of illuminating cross sections at the requisite frame rate, which is an order of magnitude (and then some) beyond what any commercial projector is capable of. Second, we have a dynamic volume - a spinning or oscillating projection surface - onto which the projector can render these cross sections. Then, in order to enable interactivity, we need a passive mechanism which will relay the image from projection volume into open space, where it may be freely touched. This gives us a dynamic holographic volume.

Infrastructure to support and integrate the two physical components is a major aspect of this project. Custom software was written to generate and display cross sections from 3D geometry. The following sections will discuss each component in detail, as well as how they are connected and integrated through custom hardware and software.

2.1.1 High Frame Rate Projector

Perhaps the most critical component of our design is the high frame rate projector. Prior work, notably that of Jones et al at USC [11], utilized projector technology based on digital micromirror devices, or DMDs, to produce images at up to 5760 Hz. A digital micromirror device is exactly what it sounds like — a board which consists of hundreds of thousands of microscopic mirrors, micrometers on a side, arranged in a fine grid. Each mirror may be individually adjusted to reflect light from an incident bulb back onto a screen for display, or away from the projection surface (and onto a heatsink) so it is not seen [3].

It is theoretically possible to switch these mirrors at rates of up to 300 kHz, enabling the projection of patterns at up to 300 thousand frames per second. Practically speaking, though, DMDs come with a number of drawbacks. First, they are very expensive. Texas Instruments' flagship model, which is to the best of our knowledge the most capable DMD on the market, retails for nearly $1000 (without the comparably pricey proprietary controller) and can only switch at 32.5 kHz. Models in our price range switch at around a sixth that speed [2]. Second, the binary switching speeds are misleading. Highly detailed monochrome (ie black and white) images may be projected at these rates, but 8-bit grayscale can be projected at only an eighth as fast, since each grayscale pixel is encoded in eight separate mirror rotations. Full color requires either a further 3x reduction in frame-rate to accommodate projection of separate red, green, and blue channels, or a 3x increase in price to purchase three independent DMDs [10].

We created instead a novel system which will effectively convert the extremely high spatial resolution of today's LCD displays into a far higher "temporal resolution", or framerate, than they would ordinarily be capable of producing. A 4K monitor is capable of displaying $3840 \times 2160 = 8294400$ pixels. A standard DVD, by contrast, reproduces content at 720x480=345600 pixels. A 4K monitor can therefore, hypothetically, show 24 DVD resolution images simultaneously on a single screen. If the monitor refreshes at an industry standard 144Hz, then in a single second it can show $144 \times 24 = 3456$ DVD resolution frames in one second. The issue, of course, is that these frames will appear on a grid of 24 virtual sub-screens, all in parallel. Somehow, we want to combine all these images so they appear in one place.

It isn't immediately obvious how this may be accomplished using a conventional monitor. Imagine instead, though, that we want to combine the images from 24 separate projectors onto a single screen. Distortion effects notwithstanding, this is a simple matter of pointing all of them at the same target. Now, as a practical matter, digital video projectors are quite expensive, and reasonably affordable commercial models only output 60Hz, which means we would need at least 50 of them. Controlling these 50 separate projectors would be a herculean task, requiring dozens of computers and an enormous amount of power, to say nothing of the physical space that they would occupy. Indeed, we would certainly be better off using the DMDs. But what if we could take our single high resolution monitor and convert it into 24 mini DVD-resolution projectors?

It turns out we can. By scaffolding optical wedge prisms in front of the display, we can redirect the light from each sub-screen onto a single projection surface. A wedge prism can take incoming light and direct it at an arbitrary angle determined by its construction. More precisely, incoming rays are deflected by an angle δ according to the relation $\delta = (n - a)a$.

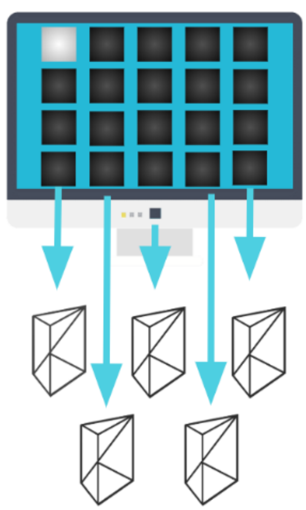


Figure 2: An array of prisms placed in front of a grid of squares

Unfortunately, simply placing a prism in front of a monitor will not turn it into a projector, as prisms do not focus light. The prisms will therefore be designed such that all the sub-screens converge on a single condenser lens, which will focus all the incoming light onto one screen. A nagging problem remains, though. As the monitor is on all at once, each of the 24 sub-screens are simultaneously illuminated, and that means that sets of 24 frames will be projected on top of each other, with the set changing 144 times per second. Instead, we want each of the 24 individual frames to flash in sequence, but to achieve this we need to be able to selectively turn one frame on at a time.

LCD screens do not produce their own light, so the image they produce is invisible to the eye if the backlight is removed. This is generally considered a shortcoming of the technology, and the principle issue OLED displays seek to solve [8]. For our purposes, however, this shortcoming is actually a very useful property. If we remove the monitor's stock backlight and replace it with a grid of 24 individually controlled lights which can selectively illuminate the area behind a single prism, then as long as we can control them with enough precision, we can coerce the monitor into projecting only a single frame at a time. LEDs can switch on and off several of orders of magnitude faster than 4000Hz, so this is not an issue.

2.1.2 Volumetric Display

With the goal of rendering a holographic image in mind, we would need a display, or a medium for light to refract upon, in order for the projected 3D object to become visible to viewers. First proposed by Luzy and Dupuis in 1912 [12], the volumetric display has been an active area of research in the field of 3D display, to present 3D volume or animated images in a given space [12].

There are two types of volumetric display, static and dynamic. Static display usually involves the use of gas and a laser for illumination, and has found its place in various entertainment venues, concerts, and festivals around the world. The major downsides are the cost involved and the technical difficulties in controlling multiple light sources to render 3D volumes [21].

Dynamic or swept-volume displays, on the other hand, make use of a rapidly moving display surface, and rely on persistence of vision to create a 3D object from a series of different perspectives. This technology has yet to be adopted widely, but shows a lot of promise as only one projector is needed to create 3D volumes. Thus, in order to render realistic 3D images on a limited budget, the use of a swept-volume display is a preferable choice to integrate with our high frame rate projector.

The simplest and most common design for a swept-volume display is a rotating mirror angled at 45 degrees, with the high speed projector directly on top [12].

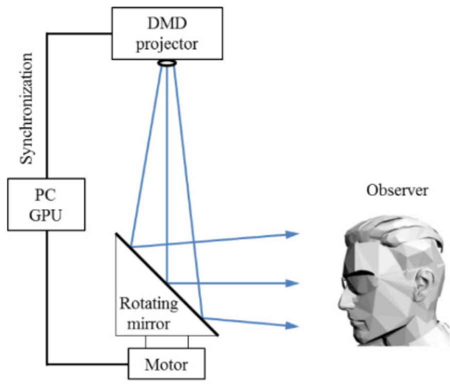


Figure 3: Swept volume display using rotating mirror [12]

The mirror will be attached to a motor that rotates persistently at at least 900 RPM in order to give a 15 Hz refresh rate, which is the sufficient rate required to generate a reliable 3D persistence of vision effect [11]. Furthermore, since we are relying on persistence of vision to display our 3D object, each 2D image that we are projecting needs to appear at the correct rotating mirror's position. In another words, we need to keep the mirror spinning at 15 Hz while constantly checking what direction the mirror is facing, to know what to display at that particular position to trick the viewers into seeing a static 3D object.

Luckily, motor control and feedback is a well-studied problem in automotive applications [5], and there are many online tutorials on integrating a DC motor equipped with hall effect sensor and dedicated driver with an off-the-shelf microcontroller to achieve what we desire.

Basically, a reading from the hall effect sensor will tell us when the motor has reached the reference position zero, where the sensor is located. Counting the number of rotations per second gives us a good estimate of the motor speed, and the reference position gives us a good estimate of where the dynamic volume is facing.

On the other hand, a motor driver is a collection of current amplifiers that takes the low current control signal from a microcontroller and convert it into a higher-current signal needed to drive the motor. Then, the motor speed can be adjusted and kept constant with pulse-width modulation.

A distinguishing feature of hologram as advocated by and depicted in many sci-fi Hollywood movies is the visible, yet intangible element. A swept-volume display only satisfies the first requirement. We want to take this

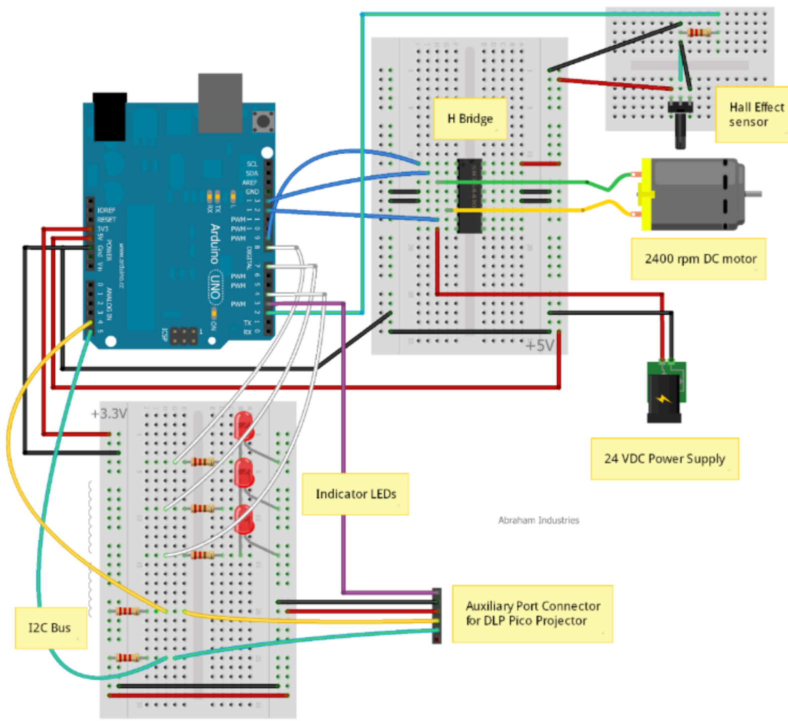


Figure 4: Sample Circuit for reading Hall Effect Sensor

a step further and introduce intangibility, to enhance users' immersion in our 3D display technology. This can be achieved with the addition of two parabolic mirrors.

Inspired by an educational toy called a mirascope, we have came up with a design that will allow our swept-volume display to function as intended while seeming to appear in mid-air.

A mirascope is a toy made of two parabolic mirrors, working together to project a small object placed at the bottom to the top, creating an optical illusion that can be described as a mirage. The working principle and physics behind the mirascope can be seen in the diagram below.

Thus, if we keep our projector setup as it is, and simply put our rotating mirror inside the mirascope, at the bottom, we will get an illusion, a mirage, or basically a holographic 3D images on the top of the mirascope, which viewers can observe and directly interact with.

The remaining challenge would be to get a mirascope large enough to contain our rotating mirror. An off-the-shelf mirascope is only 6-9 inches

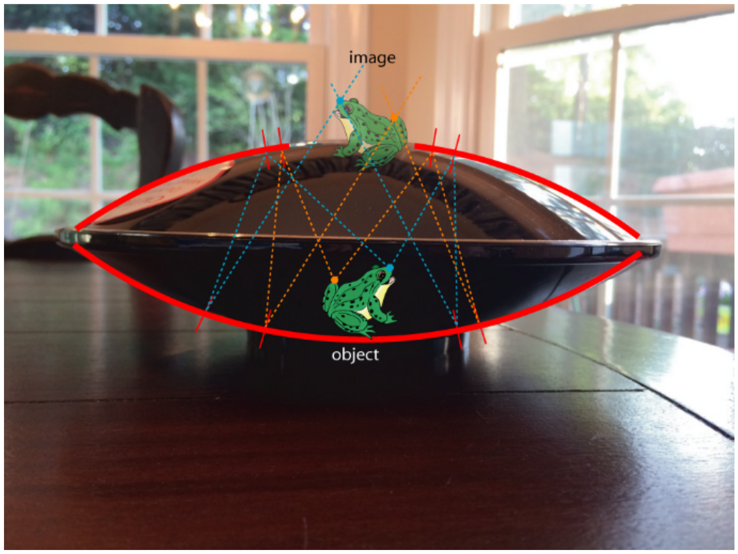


Figure 5: Working principle of a mirascope [4]

in diameter, and so we have decided to manufacture our own, to produce one that is at least 22 inches to produce a larger optical illusion for an eye-catching proof of concept and demo.

3 Infrastructure

3.1 Backlight Rig

The high frame rate projector relies on a custom designed backlight which allows for determined segments to flash in a distinct manner. In order to accomplish this, a custom housing for the backlight circuitry and the LEDs the circuitry controlled had to be designed. Our team referred to this housing as the backlight rig. Several iterations of the backlight rig were designed, as new constraints were discovered and design tradeoffs had to be considered. Let us briefly cover these design iterations in the paragraph below.

Initially, we designed a backlight rig out of plywood by hand. This design was quickly abandoned, as the tolerances of a hand assembled backlight rig were too loose for the precision required for this project. From here we looked into two types of machine based manufacturing. The next design we came up with was a model to be 3D printed. Upon consulting with the

NU 3D Printing Studio, we found it would cost upwards of \$50 to print this design. Given that this design was a prototype and would likely be changed, we opted to search for more cost effective approaches. The studio suggested using a laser cutter to cut the backlight rig out of plywood, instead of using much more expensive 3D printing filament. This required a redesign, but ultimately led to a production cost of around \$20. In the following paragraph, we will discuss the final design of the backlight rig and some of the design decisions we made.

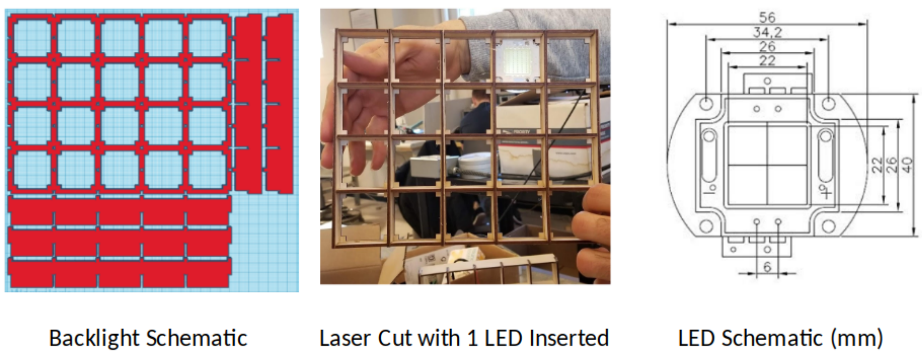


Figure 6: Backlight schematic and assembly

The final design was made using an online tool called TinkerCad. This tool allowed for easy 3D modeling with little prior knowledge, and gave us the necessary files to laser cut the plywood for assembling the backlight rig. The rig was designed in the following way. We needed to fit 20 LEDs in a rectangle shape, this was accomplished by arranging the LEDs in a grid of 5 columns and 4 rows. The center cutout to hold the LEDs was 40mm, and had 1mm padding on each side. The lips of the LEDs were cut by the machine shop, allowing them to be placed with such marginal spacing. Additionally, the corners of the LED were cut out by the manufacturer, allowing the backlight rig to have corners which latched into those holes. This structure was called a cubby, as it served as the housing for a single LED. Several of these cubbies were placed next to each other with $1/8^{\text{th}}$ inch or 3.175mm spacing between them. This allowed us to laser cut the cubby separators from $1/8^{\text{th}}$ inch plywood. These separators had to slide together so they could be assembled and glued to the backlight rig, to make one piece. This was done by making 3.175×3.175mm holes in the backlight rig in which the separators would slide in. These separators ran in the x and

y directions, and at their intersections the x separators ran underneath the y separators through 3.175×13mm rectangular holes cut in the separators. This is easier to see in the picture of the model below, which includes the backlight, three x separators, and two y separators.

The final thing to consider was the depth of the backlight rig. The closer the LEDs were to the monitor, the more intense the light would be coming from the projector. However, if the LEDs were too close to the monitor, they caused the liquid crystals in the monitor to melt. This, obviously, is not ideal, so they had to be placed at a reasonable distance from the monitor. Through trial and error, we found the ideal depth of the backlight rig to be 29.175mm which includes the 3.175mm thickness of the backlight's LED holder. Figure 6 shows our schematic and how the result turned out.

Figure 7: Light channel assembly

3.2 Light Channels

The light channels were designed to ensure the monitor image was only refracted by its corresponding prism. For example, if the light projected from Section A of the monitor reached Prisms A and B, then the resulting projected image quality would be diluted. The light channels are square cardboard tunnels that ensured the light did not escape from it's designated path. Each channel was created by cutting out four sides of the channel and taping them together. The channels were designed to fit within the light guards, which kept the channels in place (see next section). The light channels were initially prototyped using manual cutouts, but this version had gaps between the channels, allowing light to escape. The next version was designed in Adobe Illustrator and laser cut to ensure greater precision. Each channel was 18.3 cm long, the distance from the monitor to the prism array. The channels were created with thick cardboard and covered with duct tape to increase the rigidity of the structure and to ensure no light escaped.

3.3 Light Guards

The light guards were also an integral part of this project. They helped channel the light and served as bookends for the light channels. One went on the opposite side of the laser cut backlight rig, sandwiching the monitor. The other was placed before the prism holder. These used a similar design to the backlight rig, and were structurally sound, allowing us to align the different pieces of the high frequency projector without fear of breaking the components. The full design specification is detailed in the paragraph below.

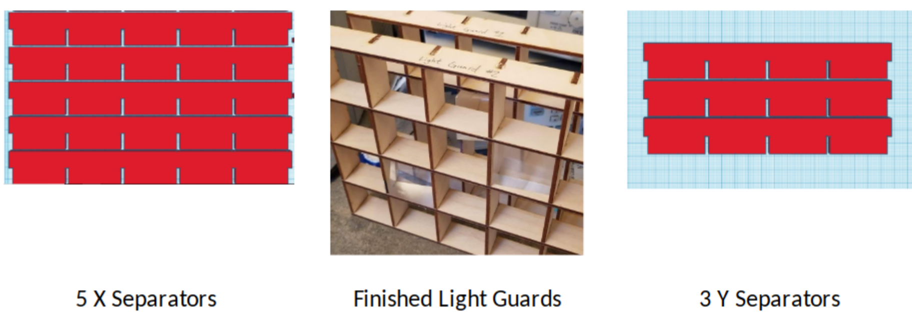


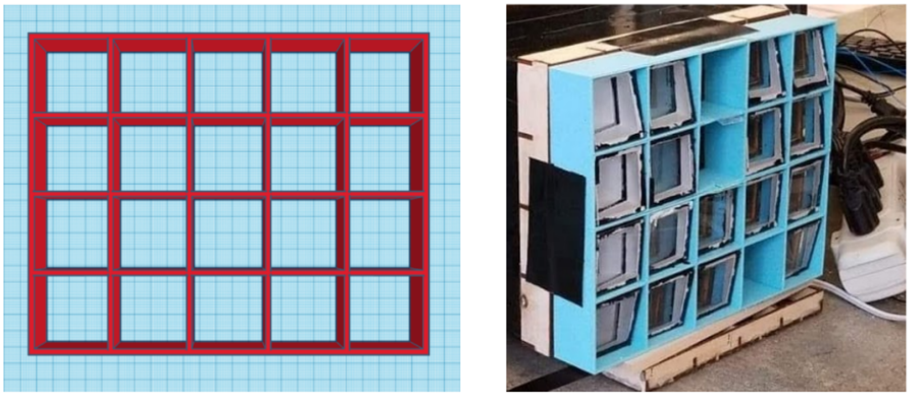
Figure 8: Light guard drafts and assembly

As previously mentioned, the light guards were designed in a similar fashion as the backlight rig. Rather than having a backlight LED holder, the light guards used only the separators discussed at length in the Backlight Rig section of this report. These separators ran the entire length and width of the backlight rig and were 26mm long. They also had cutouts at separator intersections, allowing them to slide together. In this way, the light guards essentially acted as an outline for the cubbies discussed in the Backlight Rig section of the report. Unlike the backlight rig, the light guards went through only one design iteration, as they borrowed many of the design decisions from the backlight rig. Figure 8 shows 5 x separators, the finished light guards, and 3 y separators.

3.4 Prism Holder

The prism holder was another integral part of this project's infrastructure. It held the prisms in place and allowed for their spatial adjustment for fine tuning. This went on the other side of the second light channel. Thus the light channels directed the backlight LEDs' light to the prisms. The prisms then deflected the light to a central point. Due to the imprecise sealing process of prism manufacturing, the prism holder went through many design iterations. This was designed in TinkerCad, as the backlight rig and light guards were. We know from previous sections that the cubbies were 42×42mm squares with 3.175mm spacing between them arranged in a 4×5 grid. This spacing was a constraint of the thickness of plywood we could cut utilizing NU's 3D Printing Studio. A 42mm deep prism holder was designed based on these specifications, but due to the sealing process in prism manufacturing (discussed at length in the next section of this report) the design was too small to fit the finished liquid prisms. At this point, we had already cut and designed the backlight rig and light guards, and another redesign of these components would have been costly with respect to both time and money. To solve this problem, the cubby dimensions were expanded by making the spacing between each cubby smaller. This meant that a laser cutter could not be used to manufacture this piece, as the smallest thickness of plywood we could acquire was 3.175mm. Thus, this part was made utilizing a group member's 3D printer. This printer was not previously available for manufacturing other parts of this project's infrastructure.

The final design of the prism holder expanded the cubby dimensions by 1mm on each side so that the cubbies of the prism holder were $44 \times 44mm$. This allowed us to slide the sealed liquid prisms into the cubbies without damaging them. This means that the spacing between cubbies was thinned to 1.175mm, and the overall x and y dimensions of the prism holder remained unchanged. This allowed the prism holder to match the dimensions of the light guards and the backlight rig. As previously mentioned, the prism holder was 42mm deep allowing us to adjust the prisms' placement within the holder. Pictures of the schematic, and final prism holder can be seen in Figure 9.



Prism Holder Schematic

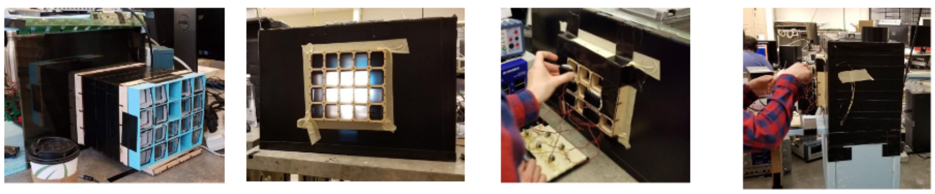
Finalized Prism Holder (Blue) Taped to 2nd

Figure 9: Prism holder schematic and assembly

3.5 Projector Chassis

The finished projector can be seen in the image below. This projector had to be housed in a chassis and lifted above the table, so that the projected light would move horizontally, hit the mirror assembly and be redirected downwards onto the dynamic volume. Additionally, this housing added to the robustness and overall sturdiness of the final project, as the whole table was rolled into the competition area from the lab. Choosing a material for this housing was difficult, as it had to be cheap, sturdy, and easy to work with given a limited toolset. The material we ended up working with was a hard styrofoam.

Using the blue hard styrofoam, we constructed a rectangular housing which centered the bottom three rows and middle three columns of the backlight rig. This specific centering strategy was used when we moved from using 20 50-watt LEDs to 5 100-watt LEDs. As discussed earlier, this design decision allowed us to get a brighter image at 1200 Hz while allowing us to use the already manufactured backlight with 20 cubbies. The depth of this rectangular housing was equal to the distance from the first light guide to the end of the prism holder. This allowed us to put the monitor on the left side and project light out of the right end of the housing. The dimensions of the rectangular housing were chosen to match the dimensions of our disassembled 240Hz monitor, 23.43in long by 13.12in wide. By propping up the high frequency projector using styrofoam and plywood, we were able to center the backlight rig in its styrofoam chassis. This chassis was then lifted up off the table with 12in tall pieces of styrofoam which ran around the perimeter of the chassis, pictures of the whole assembly can be seen below. Finally, we taped the backlight rig on the left side of the chassis, centered with the components inside and built a small shelf to hold the backlight rig's circuitry.



Finished Projector Outside Chassis | Backlight Rig Taped to Chassis | Putting in Backlight LEDs to Chassis | Finished Chassis Elevated above Table

Figure 10: Projector chassis

3.6 Mirror Assembly

As mentioned previously, the mirror assembly allowed us to project light horizontally from the high frame rate projector as the mirror redirected the light downward. This downward light was necessary to project onto our dynamic volume which faced upward due to the microscope's constraints. Additionally, due to design constraints of the high frequency projector, namely the magnification of the Fresnel lenses used to collate the light coming from the projector, the light could only travel a certain distance, 63 inches, before the light was uncollated. Due to this constraint, the light had to exit the projector, hit the mirror, and hit the projection surface within that 63 inch window. This proved to be a challenge given the 22 inch diameter of the mirascope that was used. To solve this problem, the mirror assembly had to hover above the mirascope at the proper angle.

Getting the mirror to hover above the mirascope proved to be non-trivial. We used the monitor stand, two pieces of 15 inch long hard styrofoam, the mirror, and a roll of duct tape to secure the mirror assembly in the proper place. These pieces of styrofoam were taped together to provide structural support for the mirror. The mirror was then taped to these at the proper angle. This resulting assembly was taped to the monitor stand which was left over from disassembling the 240Hz monitor. This monitor stand allowed us to position the mirror by hand through trial and error, and gave us some pivot control. Overall, this design approach allowed us to redirect light properly, even if it may have interfered with viewing angles to some degree.

Mirror Assembly with No Arm

Final Mirror Assembly with Arm

Figure 11: Mirror assembly

3.7 Dynamic Volume

3.7.1 Motor Assembly

Choosing a motor to spin the dynamic volume was a crucial part of the system as achieving a holographic illusion required the dynamic volume's rotation to be controlled, persistent, and stable. Two challenges were faced regarding the motor assembly:

1. The motor had to spin at a constant frequency of 15 Hz. Though many motors were available to us that satisfied the consistency constraint, they were simply too fast for our application and couldn't directly spin the dynamic volume. In our final implementation, we used a 12V 1000 RPM motor spinning the dynamic volume chassis directly, paired with

a L298N Dual H-Bridge Stepper Motor Controller, which satisfied all the constraints mentioned above.

2. The motor had to stay fixed to its underlying fixture, countering unwanted linear movements of the motor effectively. To address this, we 3D printed a cubical casing with a cylinder cutaway with the same dimensions of the motor. Using the extra support provided by the casing, we were able to easily stabilize the motor by duct-taping and gluing the motor assembly to the table.

Figure 12: Underside-view of the the dynamic volume chassis

3.7.2 Dynamic Volume Chassis

The dynamic volume chassis was a much less laborious process than the high frame rate projector chassis. This was due to the fewer components used in the dynamic volume and its orientation. Whereas the high frame rate projector had to be oriented horizontally and elevated above the table, the dynamic volume only had to be slightly elevated. The motor assembly was fixed to the table with duct tape. The mirascope had to be positioned about 7 inches above the table. This was done by cutting three $1 \times 6 \times 7$ inch pieces of hard blue styrofoam and using hot glue to glue them together to form a U shape. This U shape held the mirascope about 7 inches above the table. This was assembled in a U shape to allow for easy access and viewing of the motor assembly. A view of the dynamic volume chassis from a camera perpendicular to the table can be seen in Figure 12.

4 Optics

4.1 Lenses and Prisms

Modern commercial video playback devices, be they televisions, computer monitors, or projectors, all run at around 60Hz. That is, they project 60 still video frames in a single second. The highest commercially available frame rate for any type of playback device is 240Hz, a speed only found in high-end gaming monitors. For general use and even for gaming, this is, summarily, overkill. Motion will not appear more fluid or realistic at 240Hz than it will at 120Hz or even, in most cases, 60Hz. For our purposes, however, 240Hz is not sufficient. As a starting point, however, it is a useful improvement over the measly 60 we might otherwise be forced to work with.

Our full-color projector can display images at 4800Hz, a number we chose to match the grayscale performance of [1]. This is achieved by dividing a single image on the monitor's screen into a 5×4 grid of 20 individual segments, each of which shows a separate frame, and then projecting each segment through a wedge prism and a convex lens onto a common surface.

A single custom lens or prism can run $50, if not considerably more, and given we needed 20 of each, professionally manufactured custom optics were clearly out of the question. Through trial and error, we discovered that cheap Fresnel pocket magnifiers, which can be purchased in bulk for less than $0.50 each, are suitable for image focusing. This left the question of how we could custom-manufacture our own prisms with sufficient optical clarity, a question which it took us several months to answer.

As an optical component, wedge prisms are quite easy to understand. Their properties may be accurately modeled using Snell's Law. That is:

$$n_1 sin(\theta_1) = n_2 sin(\theta_2)$$

where θ is the angle between incoming light and the surface normal, and n is the index of refraction of the material through which the light is passing.

Figure 13 shows how this formula can be used design a wedge prism that deflects incoming light at the desired angle. For this application, n_1 is the index of refraction of the material from which the prism is made, and n_2 is the index of refraction of air, which is approximately 1 (the index of refraction of light in a vacuum is exactly 1).

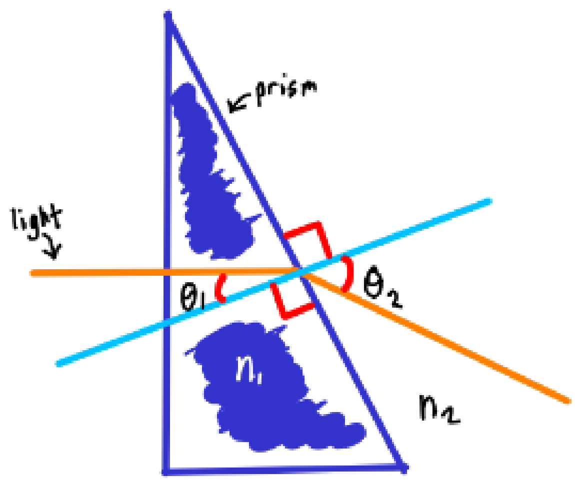


Figure 13: Snell's Law in action

Each square sector on the screen must be deflected through a slightly different angle if it is to land at precisely the same spot on the projection surface, meaning that each prism must be unique. (It is worth noting, though, that since we have chosen to place our projection surface at the geometric center of the array, some prisms will be mirror images of each other.) To calculate the required angle of deflection, we need to know how far each image must be translated in x and y, as well as the distance in z over which this deflection will occur. A shorter distance in z a shorter distance between the prism array and the screen necessitates a sharper angle of deflection, and thus larger prisms.

This distance in z is dictated by the focal length of the Fresnel lens we use. Figure 14 illustrates how a shorter focal length increases the size of the prisms. The picture on the right (blue) shows prisms designed for a lens with a 64cm focal length, the one we ultimately decided use, and the picture on the left (green) shows prisms designed for a lens with a much shorter 29.5cm focal length. It may seem surprising that we should opt for lenses with a longer focal length, given that light intensity falls off with the square of distance, but we found that the bulky prisms introduced intolerable chromatic aberration, as well as an undesirable prismatic zoom effect (a simple consequence of the fact that the hypotenuse of a right triangle is longer than either of its perpendicular sides). This chromatic aberration and zoom distortion is clearly visible in Figure 15.

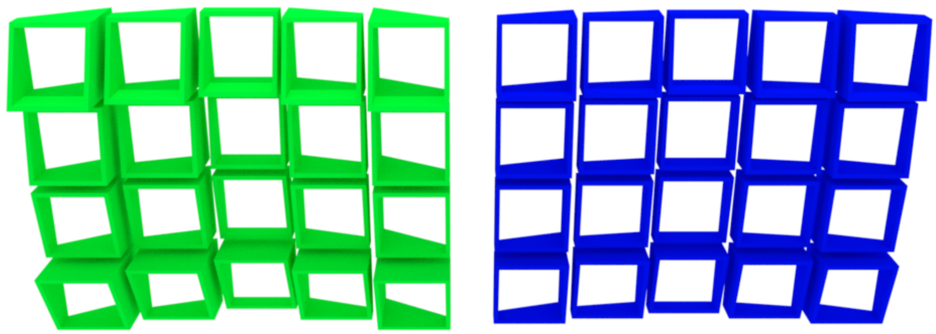


Figure 14: Prism frames. Notice how much larger the green frames, designed to deflect over just 29.5cm, are in comparison with the blue frames, designed to deflect over 64cm.

The trickiest question remains: The math behind the operation of a prism says nothing of how to actually manufacture one. Professional prisms are typically made from quartz, fused silica, or acrylic for their desirable optical qualities. We first tried 3D printing our prisms using clear resin on a stereo-lithographic printer. The ease and precision of this method was appealing, but even with extensive polishing, we found we were unable to get the necessary optical quality, and images projected through the prism appeared fuzzy and dim.

We next tried laser cutting the prisms from sheets of acrylic, but this method was even worse. Our laser cutter could only cut acrylic up to half an inch thick, and then only with very rough edges, which meant manual polishing was still necessary, and that each prism had to be assembled from six pieces. Through our trials and tribulations with the laser cutter, we did learn to effectively polish acrylic, so we thought that perhaps if a CNC machine could cut the prisms whole, we would have more luck, and indeed we did. Our CNC-cut prisms were both extremely precise and very clear, once polished, appearing almost professional to the naked eye. However, even they were not clear enough.

For our final design, we use what we have termed liquid prisms. These prisms consist of a 3D-printed frame, fitted with two acrylic windows in the front and back, and filled with mineral oil 16. These are by far the clearest prisms we were able to produce, because the thin acrylic sheets used for

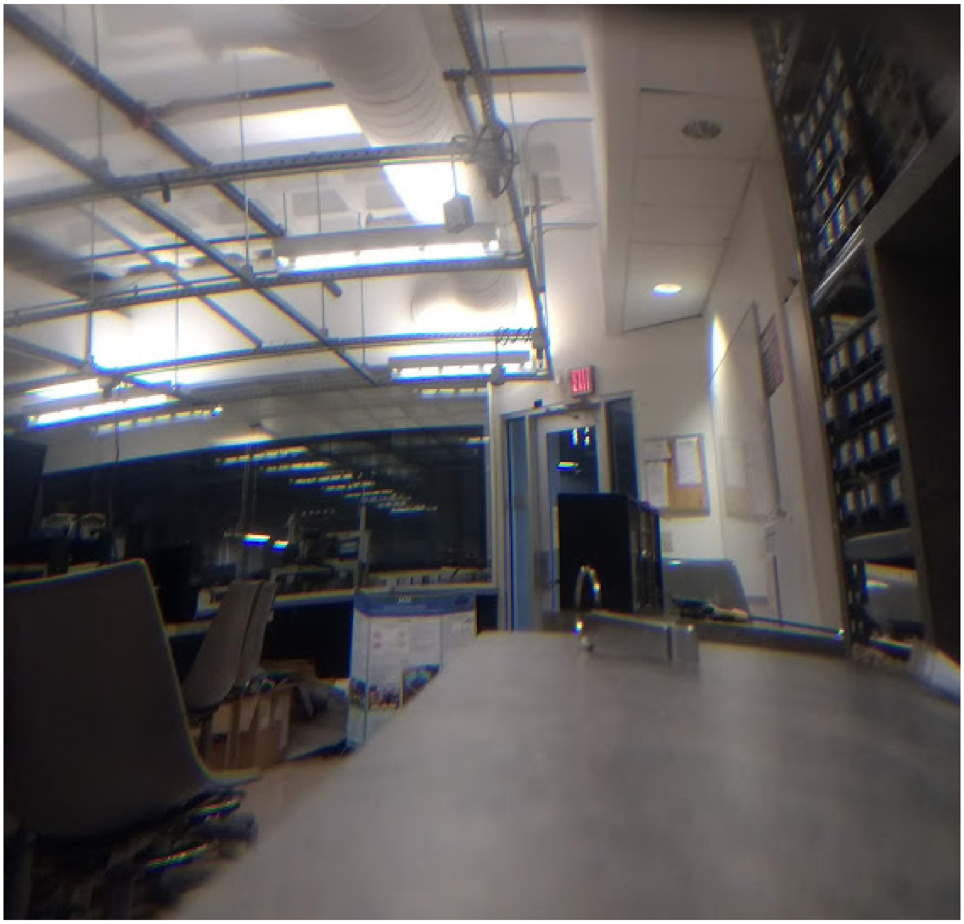


Figure 15: Zoom distortion and chromatic aberration as seen through our largest 29.5cm prism.

the windows comes professionally polished, and the mineral oil is naturally clear.

Mineral oil is a desirable liquid for this application because it is non-toxic, non-evaporative, non-reactive, and extremely clear. It has an unfortunate tendency to leak through most waterproof sealants, and is quite messy to boot, but we found it will not penetrate rubberized coatings like FlexSeal and marine epoxies like WaterWeld. Interestingly, it also has nearly the same index of refraction as acrylic, but this actually is more of a coincidence than a necessity. It follows from Snell's law that the deflection of the light

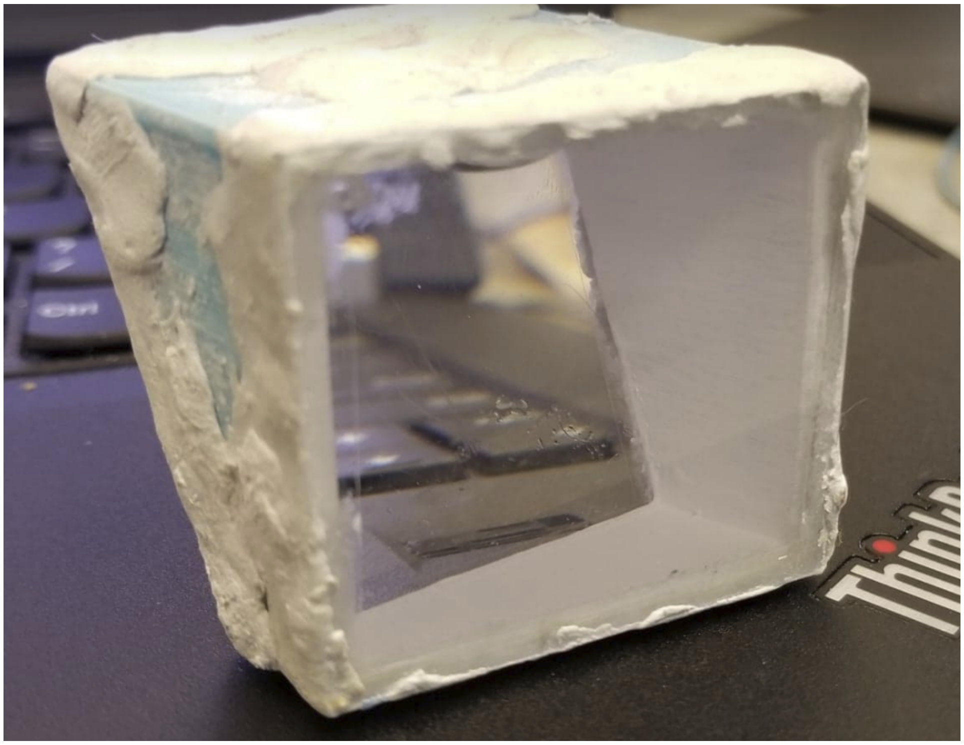


Figure 16: A fully (if somewhat crudely) assembled liquid prism.

is determined by the relative indeces of refraction of the material inside the prism and air, with the index of refraction of the window material having no effect. To see this, note that if $n_1 \sin\theta_1 = \mathrm{n}_2 \sin\theta_2 = \mathrm{n}_3 \sin\theta_3$, where n_1 is the index of refraction of the fluid inside the prism, n_2 is the index of refraction of the acrylic window, and $n_3 = 1$ is the index of refraction of air, the middle term may be removed entirely and the equality still holds. Incidentally, this explains why one cannot simply cheat and fill the cavity between the acrylic windows with air (which would have been much easier!).

The prism manufacturing process was streamlined using custom software, which accepted as input the size of each prism, the spacing between them, the dimensions of the grid in which they were to be placed, the distance in z over which the image was to be deflected, and the index of refraction of the material, and automatically generated a grid of 3D frames, which could be sent directly to our 3D printer. Figure 14 was generated using this program.

4.2 Mirror Assembly

As mentioned previously, the mirror assembly allowed us to project light horizontally from the high frame rate projector as the mirror redirected the light downward. This downward light was necessary to project onto our dynamic volume which faced upward due to the microscope's constraints. Additionally, due to design constraints of the high frequency projector, namely the magnification of the Fresnel lenses used to collate the light coming from the projector, the light could only travel a certain distance, 63 inches, before the light was uncollated. Due to this constraint, the light had to exit the projector, hit the mirror, and hit the projection surface within that 63 inch window. This proved to be a challenge given the 22 inch diameter of the mirascope that was used. To solve this problem, the mirror assembly had to hover above the mirascope at the proper angle.

Getting the mirror to hover above the mirascope proved to be non-trivial. We used the monitor stand, two pieces of 15 inch long hard styrofoam, the mirror, and a roll of duct tape to secure the mirror assembly in the proper place. These pieces of styrofoam were taped together to provide structural support for the mirror. The mirror was then taped to these at the proper angle. This resulting assembly was taped to the monitor stand which was left over from disassembling the 240Hz monitor. This monitor stand allowed us to position the mirror by hand through trial and error, and gave us some pivot control. Overall, this design approach allowed us to redirect light properly, even if it may have interfered with viewing angles to some degree.

5 Software and Control

5.1 Slicing

Generating the aforementioned cross sections is actually a very straightforward affair, thanks to the free and open source 3D-modelling suite Blender. As our projection surface is 3D printed, and in fact designed in Blender to begin with, generating the cross sections is (almost) as simple as placing the 3D model we wish to holographically project in a scene with the model of our surface, and then using Blender's "boolean modifier" tool to calculate the intersections of the two models as one is rotated relative to the other.

There are, of course, several considerations which must be made. Chief amongst them is where to place the virtual camera, so that the cross section is correctly rendered as a 2D image. The two cross sections shown in Figure 17 are in fact the very same cross section, just shown from different angles.

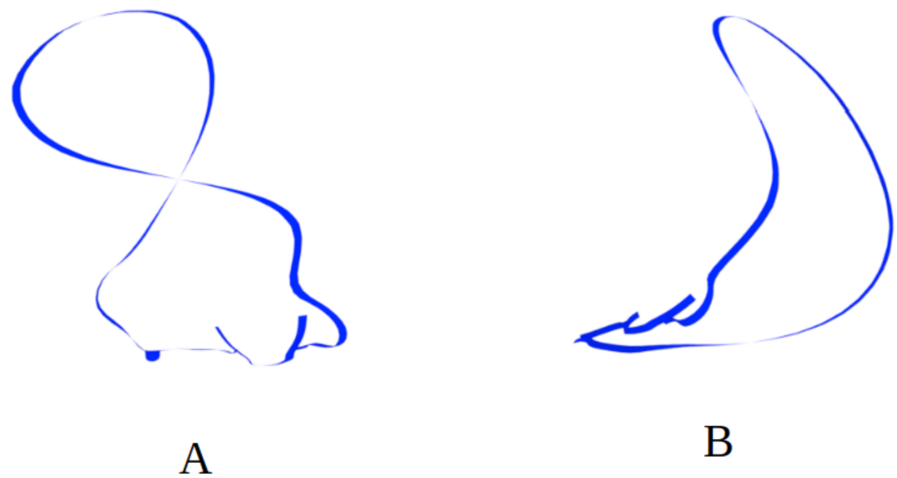


Figure 17: Top (A) and skew views of the same cross section

Figure 17A is rendered with a camera placed above the surface, and Figure 17B is produced with a camera skew to the surface. It is the configuration that produces Figure 17A, with the camera placed above the surface, which we use in our design, because (through the use of an angled mirror), our projector is effectively placed above and pointing directly down at our spinning surface.

The other important consideration, ensuring that the rendered image falls correctly on the surface, is a simple matter of adjusting the size of the rendered images, which we did by trial and error, since we could scale the rendered images up and down automatically using a simple script.

5.2 Sync

In fact, there is one other important consideration, in addition to camera placement and image scaling, which we did not touch on in the image slicing section. To form a coherent 3D illusion, each slice must be projected onto the spinning surface at precisely the right moment in time, when the surface is oriented such that it matches the rotation of the virtual surface when the image was generated. This is a critical step which we were, unfortunately, unable to complete before the competition deadline.

Synchronization is a very tricky problem because of the extremely high speeds with which we are dealing. In order for everything to line up as intended, the monitor must flash at precisely 240Hz, the motor must spin at precisely 15Hz, and the exact angular orientation of the motor must remain perfectly in sync with the monitor at all times. Even if the monitor is projecting at precisely the desired rate and the motor is spinning at exactly the correct speed, if it is even 15 degrees out of phase with the monitor, the illusion is ruined.

Motion control motors which can spin at very high speeds and maintain very precise positions exist, but they are expensive. Stepper motors are affordable, but can only spin at 60Hz or so. We are forced, therefore, to rely on a hobbyist DC motor. We successfully got it spinning at a steady 15Hz, but our method will require further tuning to ensure that it is in constant angular synchronization with the monitor. Note we cannot simply refresh the monitor to the appropriate frame when the motor spins past some set point to maintain synchronization because the monitor's 240Hz refresh rate is not sufficiently fast. Obviously, we cannot beam the motor directly to the correct angular position either. It's more a trapeze act of careful coordination.

We intend to attempt spinning the motor up to the required speed, checking if it is in phase once the speed is steady, and if not, slowing it down very slightly until the phase is correct, then rapidly speeding it back up, although we do not yet know if this algorithm will work in practice. For the purposes of demonstration, the holographic effect still works even without synchronization as long as the object being projected has perfect rotational symmetry relative to the axis of rotation of the motor. Simple shapes such as cylinders and spheres, therefore, are good candidates for projection (see Figure 18 for an example).

5.3 Display Driver

This piece of software imported the individual slices produced by the slicing software and displayed them in the Pyglet game engine at 240Hz. Due to the complicated multithreading performance tradeoffs involved with Python, this was accomplished by performing an intermediate step. This intermediate step involved stitching together the a select number of LEDs you are flashing in the backlight. The Pyglet game engine then loads these frames into memory and switches between them at a high frequency.

To begin, an input and output directory must be specified for stitching together the slices of the 3D object and saving the frames to disk. The parse_directory function searches the input directory for the slice images

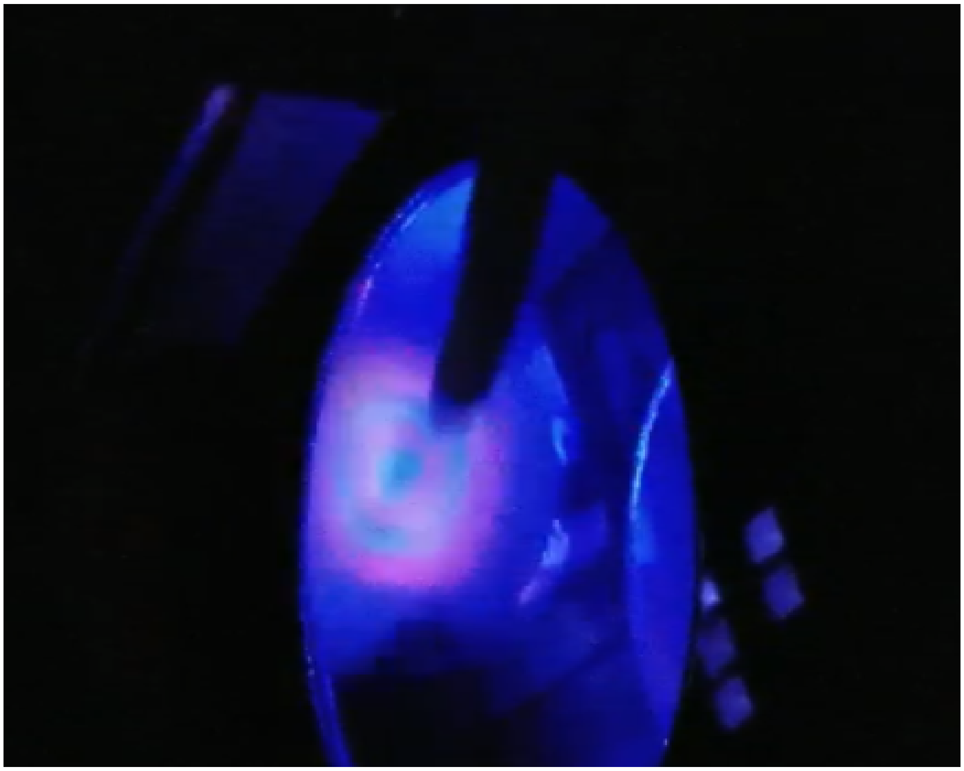


Figure 18: Actual projection of a cylinder in our final prototype.

produced by the slicing software and returns a list of lists. These sublists contain a number of slices corresponding to the number of LEDs which are flashing in the backlight. Thus, if you have 75 object slices and turn on 5 LEDs, the parse_directory function would return a list of 12 lists, each containing the filenames of 5 slices. This example matches what was used for our final design. The parse_directory function is called by the make_grid function. The make_grid function iterates through the list of lists and passes each list of image filenames to the function make_grid_helper. The function make_grid_helper takes a list of image filenames and an array of which_LEDs as well as a number of backlight array parameters (like rows, cols, box_width, box_height, and spacing in pixels) to create a frame and save it to the output directory. The which_LEDs array specifies which of the LEDs are turning on, and thus which squares need to display a slice. This program dynamically determines the position of the slices within the frame.

A monitor class takes parameters of the monitor to help create the parameters needed by the make_grid function. It's constructor takes the monitors height in pixels, width in pixels, and these measurements in inches. From here, it calculates the pixel density of a monitor with the given parameters. This allows for reuse with monitors of different sizes and aspect ratios. Finally, these values are used with the dimensions of the backlight to calculate the frame dimensions for both frame creation and the game engine.

After creating the frames from the image slices and setting up the dimensions of the game engine through the use of the Monitor class, we are ready to create a SquareImage class. This class contains a Pyglet sprite which loops through an array of loaded images. By loading the frames saved to disk from the make_grid function, this sprite can iterate through the object slices. With the sequential backlight LED flashing, we get a 1200 Hz display.

Finally, we create the Game class which tracks the frame rate of the game engine, sets the window dimensions, draws the sprite, and listens for a key press. Once the Game class is initialized and the spacebar is pressed, the game will begin to refresh the sprite as fast as possible. We were able to set the upper limit of this refreshing action to 240Hz, giving us the frame rate we desired. Finally, this Game window was relocated to the middle of the screen so it aligned with our backlight rig.

5.4 Backlight Driver

Dividing our screen into 20 separate frames is no help if all those frames are on at the same time. We want 20 separate images being projected onto the surface in even intervals over the course of 1/240th of a second, and then a separate set of 20 images being projected on the next monitor refresh, and so on. If all 20 images were illuminated simultaneously, then they would all appear as a blurry mess on the common projection surface. So, how to illuminate only part of the screen, given that sections of the screen cannot be individually turned on and off?

Fortunately, LCD panels don't produce their own light. That means that if there isn't something behind the LCD display to illuminate it, the picture it produces won't be visible, and consequently won't be projected. The backlight on standard LCD is, naturally, always illuminated, and covers the entire screen. The obvious solution, then, is to break the screen open, tear out its backlight (gingerly), and replace it with a grid of LEDs that can be individually controlled.

In our final design, we assign one 100 watt LED to each of the 20 grid squares. They are flashed in sequence, from top left to bottom right, using an Arduino. Of course, the Arduino cannot supply the requisite 100 watts directly, so they are connected across high power IRLB8721PBF MOSFETs to bench power. The Arduino simply sets the voltage across the transistor, which in turn causes the LED to turn on or off. We did attempt to use a purpose-built current driver, but only the bench-top supply could supply clean enough power.

While our projector is capable of 4800 full color frames per second, we found that for projection onto our dynamic volume, it was necessary to reduce the frame rate to 1200fps. The issue motivating that change is a consequence of a phenomenon termed PWM dimming. When LEDs are rapidly flashed, they spend only a portion of a duty cycle on. In our case, each LED was spending only $1/240^{\text{th}}$ of a second on, followed by $19/240^{\text{ths}}$ of a second off, before then flashing on again. The human eye perceives no flicker at this speed — the LED appears to be constantly on. However, its brightness appears greatly reduced, even though it is in reality achieving its full brightness for each brief pulse. We were able to mitigate this problem, while still achieving a sufficient frame rate, by reducing our projection speed to 1200Hz. This allows each LED to be on 4 times longer; 20% of the time instead of only 5%.

5.5 Motor Control

To monitor the frequency of the motor, we installed a small circular magnet on one side of the dynamic volume chassis as well as a hall effect sensor on the motor casing that enabled us to detect the magnet's presence in front of it by sending a signal back to an Arduino Nano board in charge of controlling the motor. The same Arduino controlled the pace of the motor by setting values between 0 and 255 to one of its analogue pins connected to the motor driver, with greater values corresponding to greater angular velocities.

Using the monitoring and manipulation mechanisms, the frequency of the motor was regulated every millisecond. More specifically, every millisecond, the Arduino measured the number of times the magnet has rotated. The pace of the motor was increased/decreased in proportion to the difference of the actual frequency of the motor and the ideal 15 Hz value.

6 Implementation

6.1 Timeline

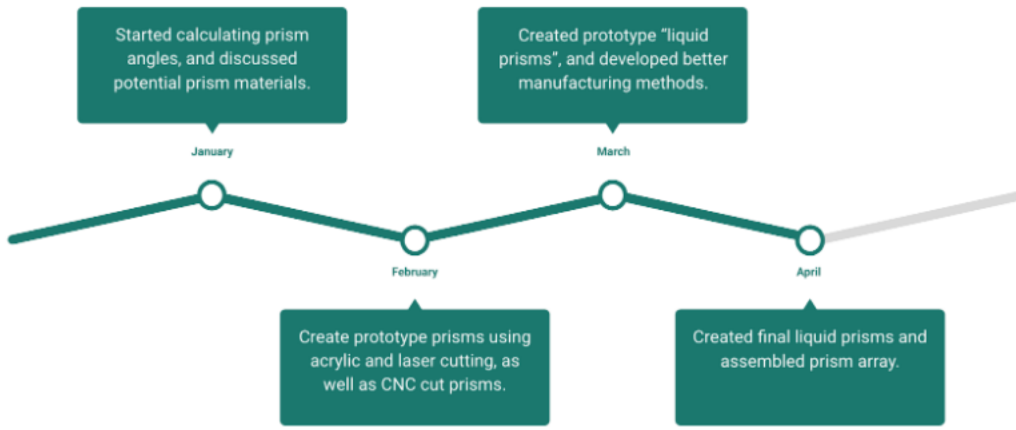


Figure 19: Timeline of prism implementation.

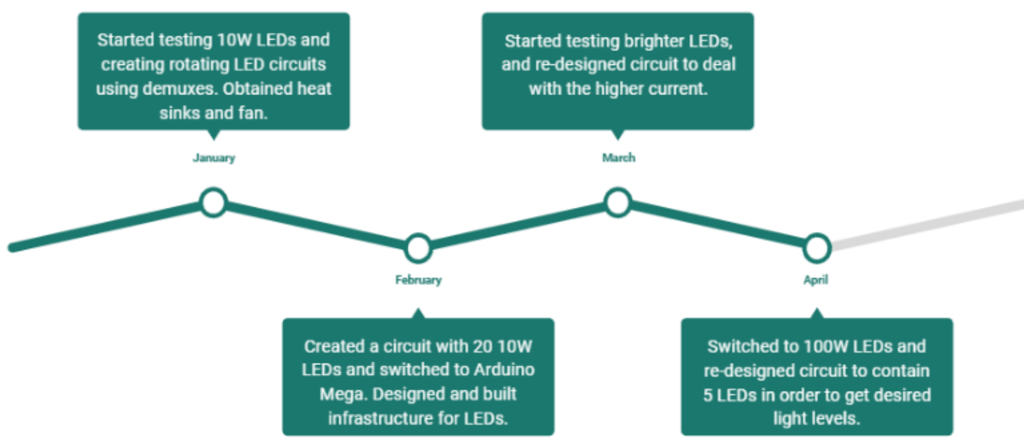


Figure 20: Timeline of Backlight Array Design and Implementation Process.

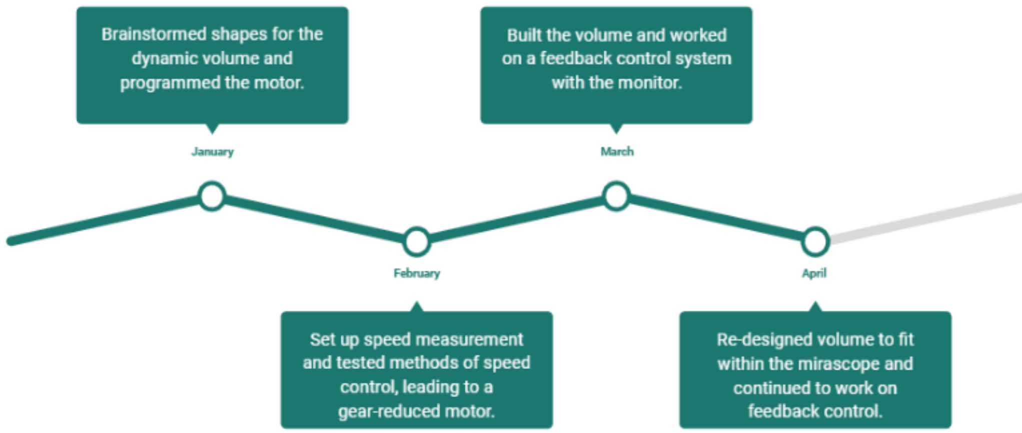


Figure 21: Timeline of Dynamic Volume Design and Implementation Process

6.2 Milestones

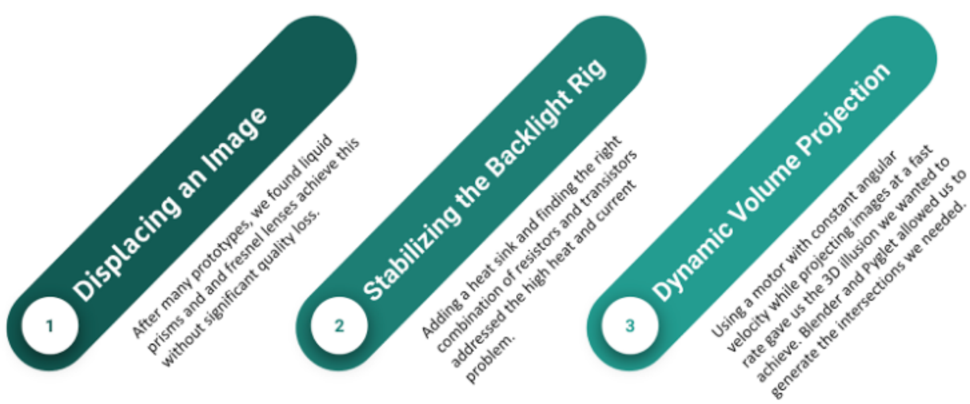


Figure 22: A roadmap of important milestones.

7 Logistics

7.1 Division of Labor

Each member of our team played a pivotal role. Luke and Quan served as project coordinators, plotting out project scope and conducting R&D. Luke focused on optics research. He spent months perfecting the process of manufacturing prisms. He also wrote all the 3D software we used. Quan worked with Sarada on the dynamic volume, focusing on motor control. Sarada and Quan also worked extensively on the electronics for the system. Josh worked hard to keep everyone in line, always being sure to point out when a particular idea seemed infeasible. His keyword was "precision." He designed most of our laser cut parts and also perfected our initial prism polishing process. Matin wrote much our control system code and was instrumental in all our physical construction processes. He's the only person on our team who can reliably disassemble a monitor without destroying it.

7.2 Expenses

The table below shows the expected cost of materials actually necessary for this project. Of course, we did a tremendous amount of experimentation, so it cost us more than this, but if another team were to start with our knowledge, this is roughly what they would need to spend to get our result. Note we're assuming access to basic equipment like breadboards and a standard PC.

Mirascope	\$290
240Hz Monitor	\$250
100 watt LEDs (x20)	\$120
3D-printed components	\$50
Laser-cut components	\$50
FreeSync GPU	\$40
Transistors (x20)	\$20
DC Motor	\$20
Fresnel Lenses (x20)	\$10
Phototransistor	\$5
Total	*\$835*

8 Conclusion

"A stable job with a stable income", "creating meaningful products", "solving the world's greatest challenges"; those are some of the most common answers as to why one would become an engineer. There is another answer that's often joked around, but it's one that should not be neglected, or brushed aside. The answer comes in many flavours, but some of its most recognizable variations are: "Making the Arc Reactor", "Creating a Time Machine", "Building a Spaceship" or "Coding the Matrix." Engineering, we argue, is not just about creating impactful products while generating stable wealth. It's also about awakening and engaging the creativity that's dormant in all of us by introducing technology so advanced, so exciting, and so indistinguishable from magic. Engineering is also about bringing science fiction closer to reality.

Our group aspires to realize hologram, since such technology has remained in science fiction for far too long. We do not claim to have accomplished the seemingly impossible. We do not claim to have successfully brought science fiction to reality. There are still kinks to be worked out, and tasks yet to be completed. However, we took great leaps, and we made great strides. We achieved the milestones that we set for ourselves, and we are closer than ever to realizing our goal. Every step that we have taken together in this journey does matter, for each step, each milestone left us with thrilling moments that awaken our inner child, and each step, each milestone will do the same to those who will see, learn or hear about our project. Our journey will encourage others to dream big, to aim high, and to rethink what's possible. Our journey will show that with Engineering, Science Fiction can and will become Reality.

9 Appendix

View all source code and project files on GitHub: `https://github.com/lhorgan/miralens`.

References

[1] 3ddisplay_uscict_siggraph2007.pdf. `http://gl.ict.usc.edu/Research/3ddisplay/3DDisplay_USCICT_SIGGRAPH2007.pdf`. (Accessed on 04/23/2019).

[2] High speed visible - products — dlp products — ti.com. http://www.ti.com/dlp-chip/advanced-light-control/high-speed-visible/products.html. (Accessed on 04/23/2019).

[3] How digital light processing (dlp) works - youtube. https://www.youtube.com/watch?v=9nb8mM3uEIc. (Accessed on 04/23/2019).

[4] Physics demonstrations: the mirascope — skulls in the stars. https://skullsinthestars.com/2016/05/02/physics-demonstrations-the-mirascope/. (Accessed on 04/23/2019).

[5] Rasterization: a practical implementation (an overview of the rasterization algorithm). https://www.scratchapixel.com/lessons/3d-basic-rendering/rasterization-practical-implementation. (Accessed on 04/23/2019).

[6] Researchers use lasers to display 'true' 3-d objects : Newscenter. https://www.rochester.edu/newscenter/researchers-use-lasers-display-true-3-d-objects/. (Accessed on 04/23/2019).

[7] Carl Berkley and Rudolf Feldt. Cathode-ray-tube applications in photography and optics. *Journal of the Society of Motion Picture Engineers*, 53(1):64–85, jul 1949.

[8] Alastair Buckley. *Organic Light-Emitting Diodes*. Woodhead Publishing, 2013.

[9] F. W. EDRIDGE-GREEN. Persistence of vision. *Nature*, 155(3928):178–178, feb 1945.

[10] Rolf R. Hainich and Oliver Bimber. *Displays: fundamentals, applications, and outlook*. A K Peters, 2011.

[11] Andrew Jones, Ian McDowall, Hideshi Yamada, Mark Bolas, and Paul Debevec. Rendering for an interactive 360 light field display. In *ACM SIGGRAPH 2007 papers on - SIGGRAPH 07*. ACM Press, 2007.

[12] Do-Hyeong Kim, Munkh-Uchral Erdenebat, Ki-Chul Kwon, Ji-Seong Jeong, Jae-Won Lee, Kyung-Ah Kim, Nam Kim, and Kwan-Hee Yoo. Real-time 3d display system based on computer-generated integral imaging technique using enhanced ISPP for hexagonal lens array. *Applied Optics*, 52(34):8411, nov 2013.

[13] Yoshihiro Watanabe, Gaku Narita, Sho Tatsuno, Takeshi Yuasa, Kiwamu Sumino, and Masatoshi Ishikawa. High-speed 8-bit image projector at 1,000 fps with 3 ms delay. 12 2015.