How to See Impossible Colors: First Steps Toward the Oz Vision Display

James Fong
Ren Ng, Ed.
Austin Roorda, Ed.

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2021-151
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2021/EECS-2021-151.html

May 21, 2021

Acknowledgement

I would like to thank Professor Ren Ng, Professor Austin Roorda, Professor Will Tuten, Dr. Alexandra Boehm, Dr. John Erik Vanston, Jay Shenoy, Rishi Upadhyay, Varsha Ramakrishnan, Yi Zong, Dr. Emma Alexander, Andrew Aikawa, Utkarsh Singhal, Pavan Tiruveedhula, Peter Manohar, and my wonderful family. This work was supported by a Hellman Faculty Fellowship and FHL Vive Center Seed Grant.

How to See Impossible Colors: First Steps Toward the Oz Vision Display

by

James Fong

A thesis submitted in partial satisfaction of the

requirements for the degree of

Masters of Science

in

Electrical Engineering and Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Ren Ng, Chair
Professor Austin Roorda

Spring 2021

How to See Impossible Colors: First Steps Toward the Oz Vision Display

by James Fong

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:

Professor Ren Ng
Research Advisor

May 21, 2021

(Date)

* * * * * * *

Professor Austin Roorda
Second Reader

May 21, 2021

(Date)

How to See Impossible Colors: First Steps Toward the Oz Vision Display

Abstract

How to See Impossible Colors: First Steps Toward the Oz Vision Display

by

James Fong

Masters of Science in Electrical Engineering and Computer Science

University of California, Berkeley

Professor Ren Ng, Chair

A color is "impossible" if there is no light spectra which can produce it. To see an impossible color would be to experience an entirely new perception, not unlike a colorblind individual seeing color for the first time. Under the assumption that color is a function of a light's spectral power distribution, it is absurd to consider seeing impossible colors. However, this assumption breaks down at the cellular level. We should be able to reproduce the entire range of human visual perception by directly modifying the activity levels of light-sensitive cells in the eye. In this thesis, we exploit the eye's cellular structure and achieve multiple color percepts from a single wavelength of laser light. This goes a step beyond traditional displays, which exploit only the spectral basis behind color. If impossible colors exist, this advancement could actually *expand* the range of human visual perception, not only just *reproduce* it. We call this technology an "Oz Vision" display, in reference to a fictional city filled with brilliant colors which cannot be seen anywhere else. Building this Oz Vision display is a significant engineering challenge. Firstly, it must non-invasively work around the eye's numerous physiological barriers in order to access the individual cells responsible for color vision. Secondly, as with any ordinary display, the Oz Vision display must be easily controllable by an end user to present a variety of graphics within the new perceptual space it offers. We achieve both goals through Wizard: our newly developed software control layer built on top of an adaptive optics scanning laser ophthalmoscope (AOSLO). We present Wizard alongside key preliminary experimental results that validate the software's correctness and demonstrate the potential for exploiting the *spatial* characteristics of color. We anticipate that this work will establish a new frontier in display technology and visual psychophysics research.

To my family

Contents

List of Figures

Acknowledgments

I would like to thank my advisor, Professor Ren Ng, for introducing me into the world of research. His guidance and encouragement have brought me to new places I never thought possible. Whenever I felt lost, asking Prof. Ng for help has always been the right answer.

Thank you Professor Austin Roorda and Professor Will Tuten. Their enthusiasm for their field and kindness toward me made me feel right at home in the Vision Science community. Prof. Roorda's approach to research and attention to detail has given me an invaluable new perspective on science. Prof. Tuten's judgement calls and quick thinking on the lab floor made our experiment days possible in more ways than I can list here.

I also would like to thank Dr. Alexandra Boehm and Dr. John Erik Vanston for their help and determination to bring key experimental results across the finish line. They made the science fun and exciting even during the most confusing moments.

Thank you Jay Shenoy and Rishi Upadhyay. We all learned so much from each other on our journey into research. Although I could not fit their work into this thesis, I am truly grateful to have them as teammates.

Special thanks to Varsha Ramakrishnan, Yi Zong, Dr. Emma Alexander, Andrew Aikawa, and Utkarsh Singhal for all the wonderful discussions and fun moments we had together. Without all your help, the Oz Vision project would be massively behind where it is now.

I also would like to thank Pavan Tiruveedhula, who has an incredible knack for making things work. The whole lab knows that if something is malfunctioning, he is ready to save the day.

I would like to thank Peter Manohar. A while back, I asked him what he was working on, and he told me he was safely shooting lasers into peoples' eyes for science. I asked him how I could get involved.

This work was supported by a Hellman Faculty Fellowship and FHL Vive Center Seed Grant.

Finally, I am eternally grateful to my family who have continually supported me in all of my endeavors. They have been an infinite source of comfort and reassurance, and that has given me the confidence to dream big.

Chapter 1

Introduction to Oz Vision

The idea of Oz Vision is that we should be able to reproduce every visual percept by accurately reproducing the activity levels of individual light-sensitive cells, called *photoreceptors*, in the eye. This is opposed to accurately reproducing the patterns of light arriving at the eye's outer surface. Using a laser, we can send pulses of light, or *microdoses*, to individual photoreceptors to activate them. To know what to send to the photoreceptors, we must build up a model for how the eye normally converts photon energy distributions into neural activity. With this model, we can determine the photoreceptor activity levels produced by a target image, and then work backwards to find how to achieve those same activity levels with our laser. We hypothesize that a subject experiencing these activity levels will see the target image, despite it being produced by a single-wavelength laser.

To quantify how well we can reproduce color perception, we use *chromaticity diagrams*. These diagrams visualize photoreceptor activity levels as points in 2D space. Using these diagrams, we compactly represent the range of color perceptions, or *gamut*, achievable with a given display. This is especially relevant for visualizing the opportunity for impossible colors. An ideal Oz Vision display would have a larger gamut than what was previously thought possible, enclosing points in uncharted regions of these chromaticity diagrams. A preview of this is shown in Figure 1.1. Another key hypothesis of Oz Vision is that these novel photoreceptor activities would produce perceptions of impossible color.

This chapter goes into detail on how we model the photoreceptors' responses to a given photon distribution, as well as describe where there is the opportunity for impossible colors.

1.1 How Shape and Color are Entangled by the Eye

The purpose of the eye is to provide a rich neural signal to the brain such that it may infer the forms and material properties of its environment at a distance. The information in this neural signal is highly tuned to the downstream task of powering inferences, and is far from a perfect record of the image arriving at the eye. This is unlike a traditional camera, which optimizes the clarity and resolution of the images it captures. The physical mechanics of

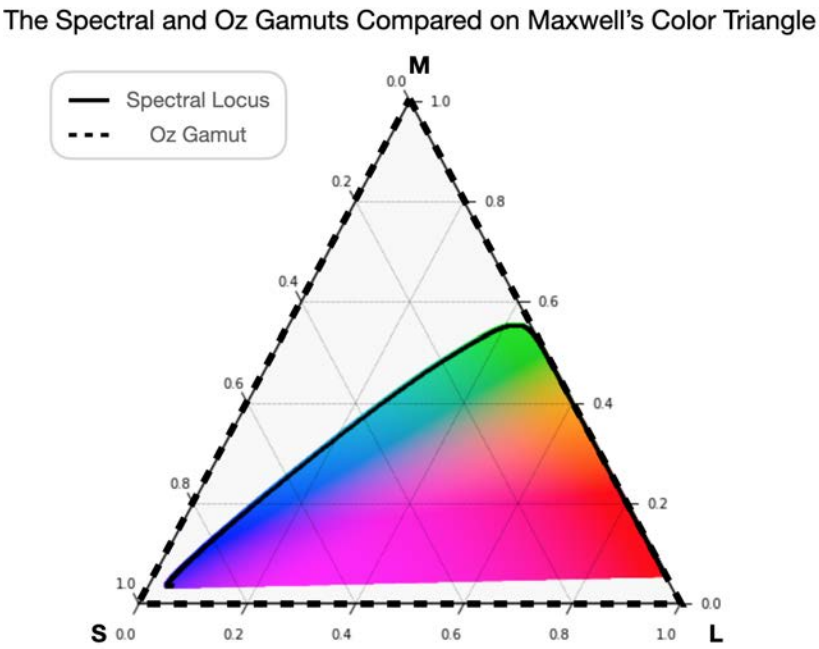

Figure 1.1: The spectral gamut visualized on Maxwell's color triangle [9]. (See Appendix A.1 for construction.) The spectral gamut is the range of photoreceptor activities achievable with a spatially uniform light source of a fixed spectral power distribution. The colored region gives a rough impression of the color percept at each chromaticity within the spectral gamut. The top edge of the spectral gamut is bounded by a curve representing all single-wavelength light sources (the *spectral locus*). The uncolored regions represent the gamut of an Oz Vision display, and contain novel photoreceptor activity levels. We hypothesize that those activity levels are perceived as impossible colors.

how the eye captures its images are thus remarkably different from that of a camera. In particular, the eye makes use of an unusual sampling strategy which confounds shape and color. In this section, we will go into detail on this sampling strategy, because it is this characteristic of the eye which we will later exploit for Oz Vision.

Before light can be sensed by the eye, it must first be focused into an image which the photoreceptors can sample. Light arriving at the eye is focused by multiple optical elements. In order of the light's arrival, they are the cornea, the anterior chamber, the iris/pupil, and the crystalline lens [2]. Although each element individually contributes to the focusing of light, for our purposes we only need to reason about their aggregate behavior. These optical elements produce a much blurrier image than what is theoretically possible. This blur actually improves the quality of the neural signal, since a low-pass filter will attenuate high frequencies that would otherwise cause aliasing by the discrete photoreceptors [24]. However, this blurring poses a challenge for Oz Vision, which aims to isolate individual

photoreceptor cells for stimulation. The eye's optics focus this slightly blurred image onto its rear interior surface, called the *retina*, which is home to the photoreceptors.

The retina is the site where the focused image is discretely sampled by a mosaic of photoreceptor cells. These photoreceptors can be divided into two main groups, the *rods* and the *cones*. The rods are best tuned for nighttime (scotopic) conditions, whereas the cones are best tuned for daytime (photopic) conditions. The rods and cones tile the photoreceptor mosaic, with approximately 92 million rods and 4.6 million cones on average per eye [5]. Although the rods are more numerous, the cones are the photoreceptors responsible for color vision and will be our primary focus. In the *fovea*, the densest region of the eye, there are over 15,000 cones per square degree[1] [5]. It is this dense tiling which enables the spatial structure of the focused image to be captured by these samples.

Each cone cell responds proportionately to the amount of light it receives, effectively sampling the image at its location. We model a cone cell's activity level as the rate of discrete "responses" that it has in reaction to a given distribution of photons at its aperture. That is, for an incoming photon of a given wavelength and location relative to the cone cell, there is a probability that it will elicit a response.[2] There are two important consequences of this model. Firstly, the activity level is linear with respect to a light source's power in Watts, since power is linear with respect to photon flux. Secondly, once a photon elicits a response, all information about that photon is lost. This means that if two different photon distributions achieve the same response rate, there is no way for the cell to differentiate between them. This is known as the principle of univariance [18, 3], and will form the basis of Oz Vision: we do not need to reproduce the photon distribution responsible for a given color percept, only the activity levels.

The cone's finite aperture is the first factor in the probability of a response to a photon. The probability of a response is highest at the center of the aperture, and decreases with distance. Since the wavelengths of visible light are comparable to the size of the cells themselves, this probability can be modeled with a 2D Gaussian [13]. The aperture can thus be visualized as a transparent region in an otherwise opaque screen, where the transparency has a Gaussian fall-off around its center. The full width at half maximum of this Gaussian is approximately half the diameter of the cone's inner segment [13], which is believed to be the cone's light-collecting organelle. This Gaussian aperture acts like an additional low-pass filter on top of that imposed by the eye's imperfect optics.

[1] This estimate is adapted from Curcio *et al.*'s originally stated 199,000 cones per mm^2, using the conversion factor of 0.0795 deg^2 per mm^2 [23, 5]. The cones become larger and less densely packed with increasing distance from the fovea [12]. In the literature, it is common to measure lengths in terms of visual angle rather than a physical quantity. This is done because of the fundamental ambiguity between size and distance for objects visible in an image. For our work, this approach is doubly useful because a degree of visual angle from the perspective of the imaging system is equal to a degree of visual angle from the perspective of the subject being imaged.

[2] The exact choice of what constitutes a "response" is mostly just a convention. One popular choice for a "response" is a single photoisomerization event, which is when a light-sensitive protein in the cone cell changes shape in response to a photon. Ultimately, we will only care if the response rates for two different light sources are equal, independent of whatever those responses are made of.

The cone's sensitivity to different wavelengths is the second factor in the probability of a response to a photon. The probability of a response is maximized at some cone-dependent wavelength, and decreases smoothly for longer and shorter wavelengths. This probability as a function of wavelength is known as a cone's quantal sensitivity. If we divide the quantal sensitivity by the energy of a photon at each wavelength, we arrive at the cone fundamental. This fundamental gives the normalized activity level of a cone for a single-wavelength light source of unit energy, and completely defines that cone's behavior for different light spectra.[3] In humans with normal color vision, there are three different fundamentals which peak at relatively long (L), medium (M), and short (S) wavelengths. A cone is therefore called an "L," "M," or "S" cone depending on which of the three fundamentals it expresses. These fundamentals are shown in Figure 1.2.

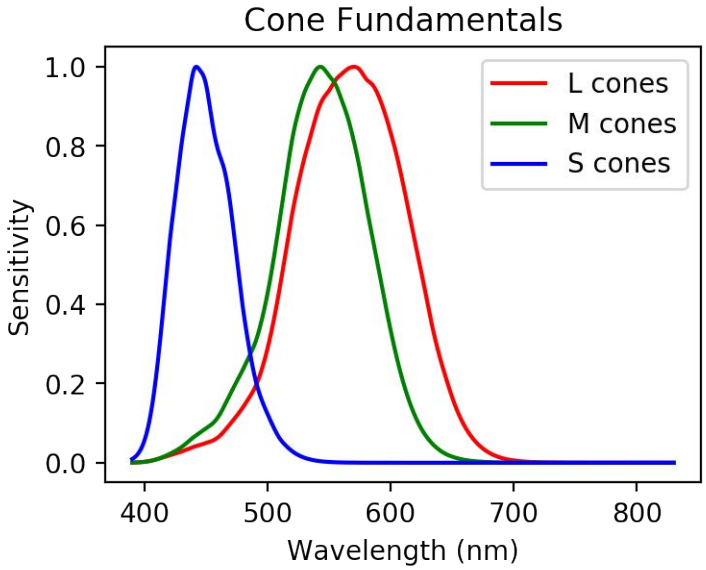

Figure 1.2: The cone fundamentals for each of the LMS cone types. The probability that a photon elicits a response from a photoreceptor depends on the photon's wavelength. The cone fundamentals are an arbitrary scaling of those probabilities divided by the photon energy. The arbitrary scaling exists because the absolute activity levels are difficult to measure, and for many color calculations only the relative values matter. In this paper we use the common convention to scale these fundamentals to fill the full 0 to 1 range. This value is also known as the "sensitivity." These fundamentals are provided by Stockman *et al.* in [22].

Since each cone cell entangles spatial structure and color into only a single activity level,

[3]Differences in pigment density in different parts of the eye mean that the exact shape of the fundamentals depends on the size of the stimulus on the retina. The fundamentals here are for a stimulus subtending a visual angle of 2 degrees.

there are cases where the shape and color become ambiguous. The material covered so far is shown in Figure 1.3, along with an example of such an ambiguous case. This ambiguity between spatial structure and spectral structure is the underlying principle of Oz Vision.

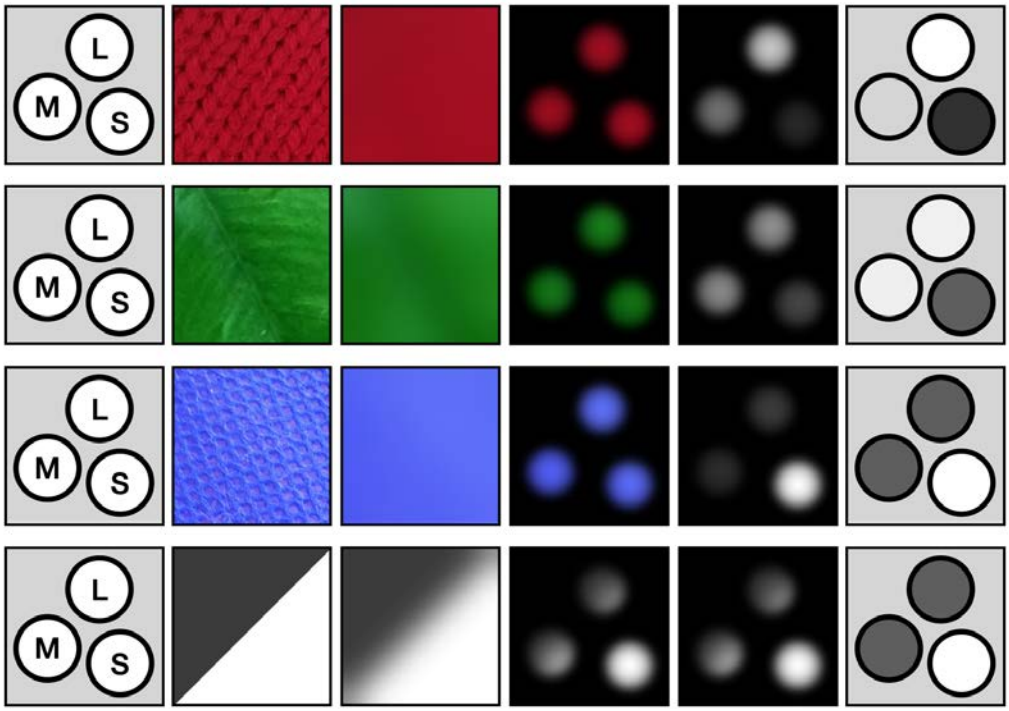

Figure 1.3: Overview of how different image patches elicit responses in individual photoreceptors. The columns, from left to right: 1) A toy photoreceptor mosaic, consisting of one photoreceptor of each type. 2) An incoming image from the environment, as it would appear without any blur. 3) The image blurred by the eye's imperfect optics. 4) The image masked by each photoreceptor's Gaussian aperture. 5) each cone integrates across the spectral dimension, with each wavelength weighted by its fundamental. 6) Each cone integrates across its aperture, producing a 1D activity level. Even though each photoreceptor is individually colorblind, by comparing the activity levels across LMS types, the visual system can discriminate between each patch. This is true except for the last two rows. In that case, the same LMS activity levels can be equally explained by a uniform blue image or a sharp luminance edge. This is how spatial structure and color information are entangled.[4]

[4]The eye can resolve this entanglement by shifting its gaze and resampling the image at different locations. This eye motion is later covered in Section 2.1.

1.2 Locating Impossible Colors

The sensation of color arises from the relative activation levels among the LMS cone cells, but the shape of the cone fundamentals prevents some relative activations from occurring. We hypothesize that these "impossible" relative activations will be perceived as impossible colors. In this section we will introduce methods for visualizing relative activity levels as points in 3D and 2D spaces, which will make apparent the unexplored regions where impossible colors might live.

Using the cone fundamentals, we can compute the relative activity levels for the L, M, and S cones for a spatially uniform light source arriving at the retina. Let the light's spectral power distribution (SPD) be $\Phi(\lambda)$, and let the cone fundamentals be $F_L(\lambda)$, $F_M(\lambda)$, and $F_S(\lambda)$. We compute the LMS activity as follows:

$$L = \int F_L(\lambda)\Phi(\lambda)d\lambda$$

$$M = \int F_M(\lambda)\Phi(\lambda)d\lambda$$

$$S = \int F_S(\lambda)\Phi(\lambda)d\lambda$$

To see the structure of this transformation, we will first examine how simple single-wavelength light sources of unit energy map to tristimulus values in LMS space. Such a light source with wavelength ω has a unit impulse at ω for its SPD:

$$L_\omega = \int F_L(\lambda)\delta(\lambda-\omega)d\lambda = F_L(\omega)$$

$$M_\omega = \int F_M(\lambda)\delta(\lambda-\omega)d\lambda = F_M(\omega)$$

$$S_\omega = \int F_S(\lambda)\delta(\lambda-\omega)d\lambda = F_S(\omega)$$

By plotting this tristimulus value as a function of ω, we get a curve called the *spectral locus*. This curve is shown in Figure 1.4.

The spectral locus defines the extent of colors achievable for any uniform light source. This is because any SPD is really just a mixture of single-wavelength sources. Since the transformation to LMS space is linear, the corresponding tristimulus value in LMS space is also just a mixture of tristimulus values along the spectral locus. Put precisely, any point in LMS space is achievable if it can be made by adding together positively scaled points on the spectral locus. We will name the reachable volume of LMS space the *spectral gamut*.

To visualize this volume, it is common to flatten the 3D LMS space down to a 2D space. Importantly, we can do this while approximately preserving the colors of the 3D points. This is because the color perceived from a given LMS point does not change significantly if we scale the LMS components by some positive constant. By removing this degree of freedom,

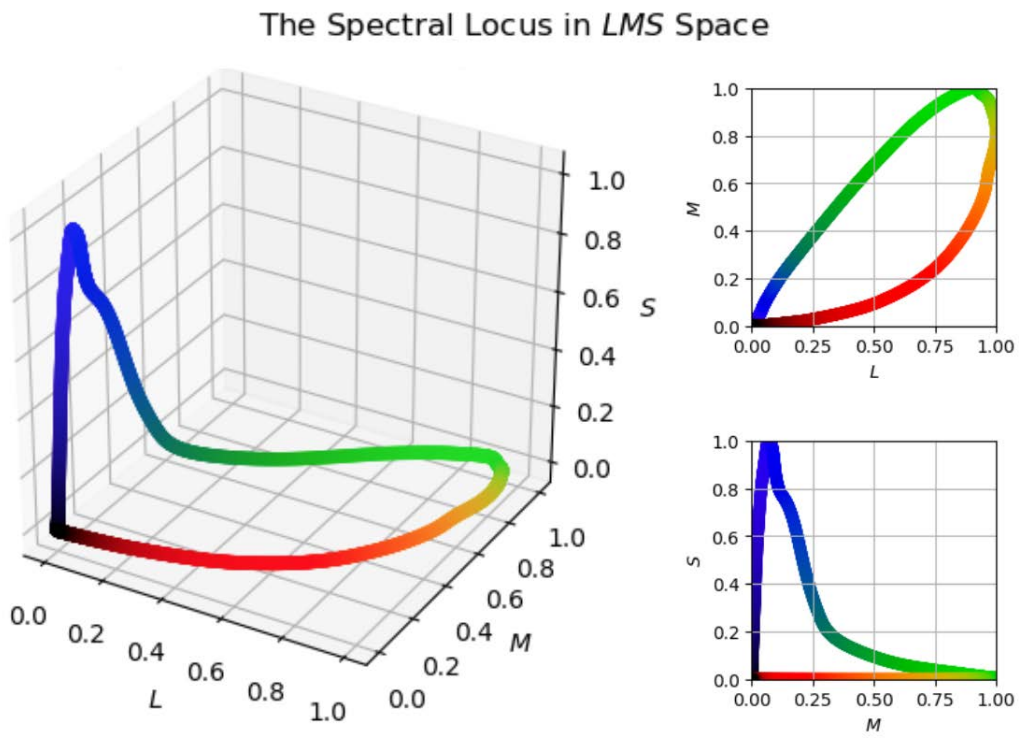

Figure 1.4: The spectral locus viewed in LMS space, derived from the cone fundamentals F_L, F_M, and F_S. Left: an orthographic projection of the curve. Right: a top-down and side view of the same curve. The color along the curve shown here is intended to give a rough idea of the color perceived at each wavelength.

we arrive at a 2D space with all of the colors intact. The resulting 2D coordinates are called *chromaticities*.

The potential for impossible colors exists in the regions of LMS space outside of the spectral gamut. This is visualized in two different chromaticity spaces, in Figures 1.1 and 1.5. Each figure depicts the spectral gamut in approximate color embedded within a larger "Oz gamut" which represents the full range of LMS values. The Oz gamut contains many novel activity levels, such as the "pure" L/M/S points for which only a single cone class is activated. Mixing these pure points yields the entire Oz gamut. Also shown is equal energy white (EEW) which is the color for $\Phi(\lambda) = 1$. The derivations of these spaces are given in appendix sections A.1, A.2.

When referring to points in a particular space, we will use vector notation following the name of the space. For example, $\texttt{LMS}\begin{bmatrix}1 & 2 & 3.4\end{bmatrix}$ refers to a joint activity level with L cones at activity level 1, M cones at level 2, and S cones at level 3.4. Similarly, $\texttt{xy}\begin{bmatrix}-0.12 & 0.34\end{bmatrix}$ is the point in the modified CIE xy space with x at -0.12 and y at 0.34.

Our research requires a distinction between the perceived *colors* and the numerical *tris-*

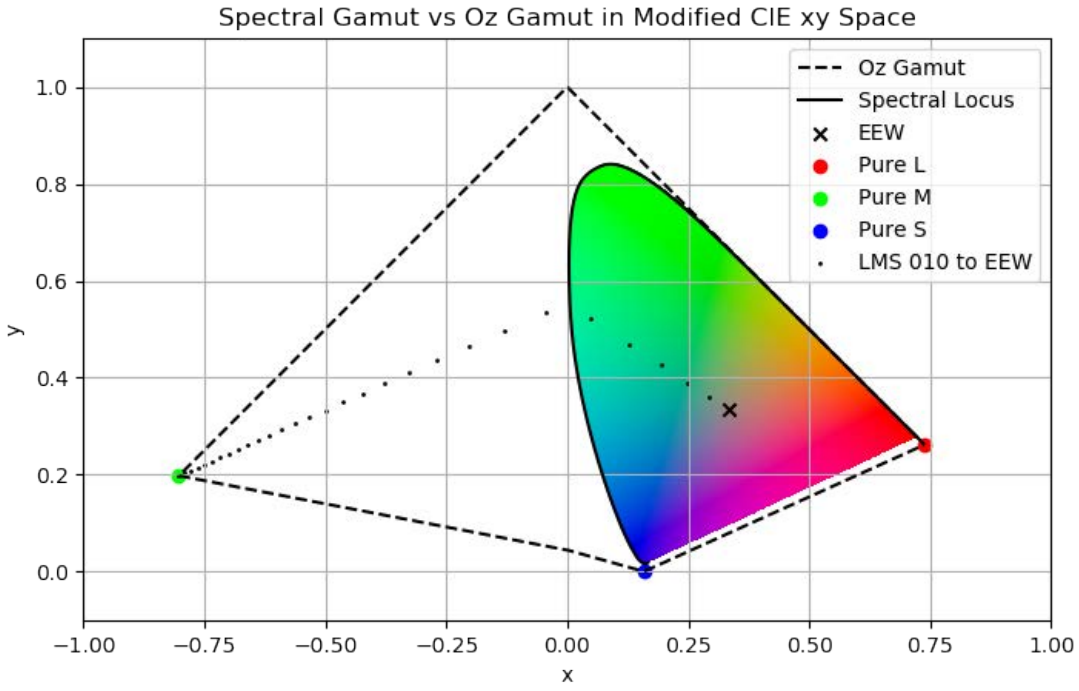

Figure 1.5: The spectral gamut visualized in Modified CIE xy space. (See Appendix A.2 for construction.) The colored region gives a rough impression of the color percept at each chromaticity in the spectral gamut. The dashed region represents a hypothetical "Oz gamut" which includes all points in LMS space. The modification to standard CIE xy causes ordinarily straight lines to bend crossing the y axis. Pictured is a series of 30 points along a straight path connecting pure M and EEW in LMS space.

timulus values and *chromaticities*. Tristimulus values and chromaticities are quantitative descriptions of how active the L/M/S cones are. A color is the qualitative perception produced from this L/M/S activity. We distinguish between the numerical values and the perception because we do not yet know how points outside the spectral gamut will be perceived. We expect that such points elicit impossible colors, but there is a chance they might not.

1.3 Hacking Human Vision

As previously shown, the sampling strategy of the eye creates ambiguity between spatial and spectral structures, and yet it also does not make full use of LMS space. These two properties raise the question: could this ambiguity be exploited in order to expand the range of LMS outputs we can achieve? If so, then these outputs could correspond to chromaticities

of impossible colors. This section provides a high-level explanation for how we might achieve this.

As shown in Figure 1.3, it is possible to use spatial structure to create exactly the same LMS activity as for a given color. We can take this idea further, and simply send targeted microdoses to individual cone cells. This idea is shown in Figure 1.6. Using this method, we can achieve all three pure tristimulus values, each activating only a single cone class. Thus, by making mixtures of these three pure tristimulus values, we arrive at the Oz gamut, which spans the entire LMS space.

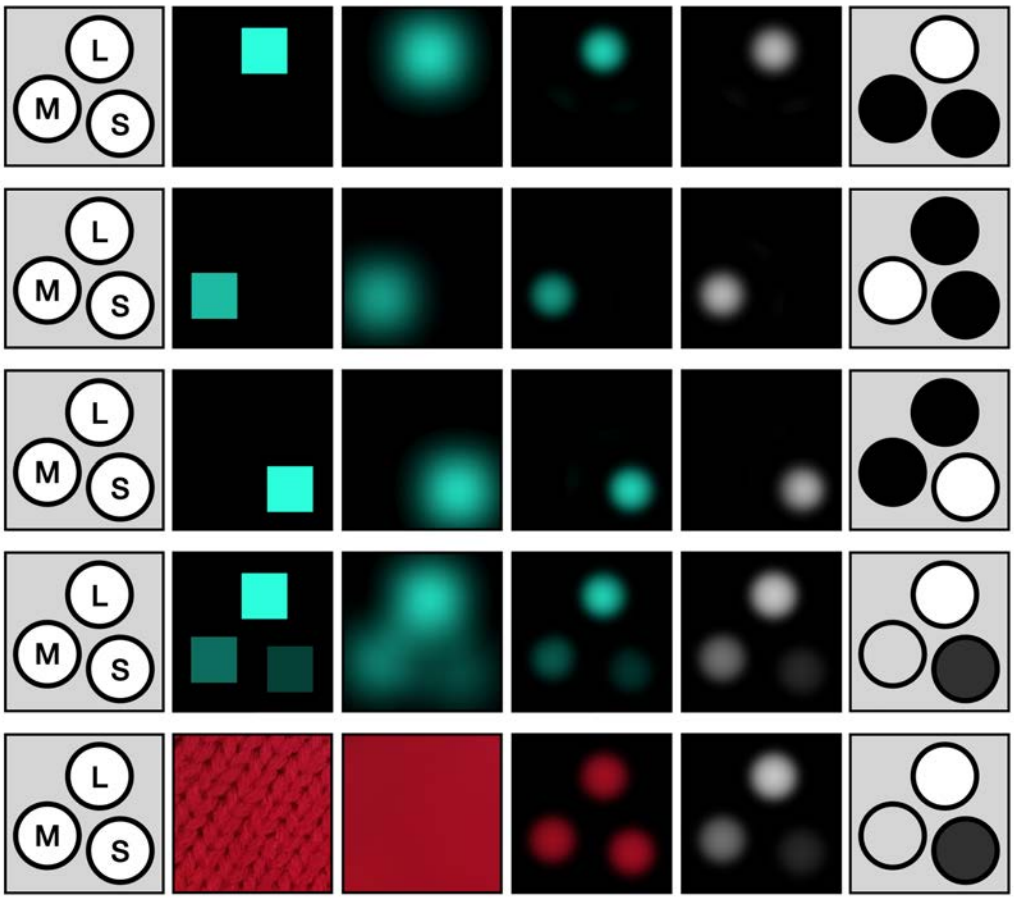

Figure 1.6: Overview of how we can use precise targeting of 490 nm light to directly modify LMS activations. The columns are exactly as in Figure 1.3. The first three rows each show a stimulus which activates only a single L/M/S cone. These images form a linear basis which can activate any desired tristimulus value on this retina. By the principle of univariance, this level of control is sufficient to elicit every possible color percept. The last two rows show an example of this, where we use this 490 nm light to create a red percept.

Achieving these Oz presentations requires hardware with capabilities far beyond what is found in a consumer display:

- It must capture an image of the subject's cone mosaic in order to know where to send the microdoses.
- It must deliver these microdoses with sufficient precision. If the light is sent to the wrong location, incorrect cones will be stimulated. If the light is not focused well enough, neighboring cones will be stimulated.

For our Oz Vision display, we use an adaptive optics scanning laser ophthalmoscope (AOSLO) which offers both capabilities. The full technical details of its construction are provided in the literature [8, 16], but here we provide a brief overview. The AOSLO is a confocal, diffraction-limited retinal imaging system capable of recording a live video feed of the subject's photoreceptor mosaic. It uses adaptive optics to concentrate a beam of laser light onto the subject's retina at a target location. This light reflects off of the retinal tissue and returns through the same optical pathway as it entered. A beamsplitter reflects the returning light into a photomultiplier tube (PMT), which then gives a record of the reflectivity of the retina at the target location [19]. To acquire an image, we scan the laser's target location over the retina across a 1×1 degree field-of-view. This scan pattern is similar to that of a cathode ray tube (CRT), with a slow vertical scan and a fast horizontal scan. Individual PMT spatial samples are then assembled into a set of 512×512 frames at ≈ 30 fps, and are sent to a consumer desktop computer using a custom-built FPGA. An example set of these frames is shown in Figure 1.7.

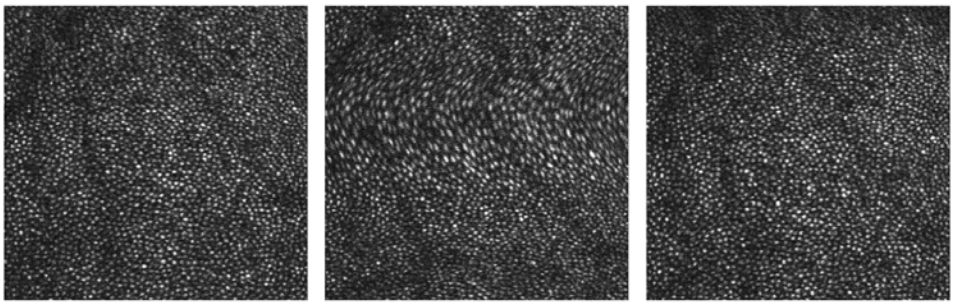

Figure 1.7: A sequence of three AOSLO frames of the cone mosaic. Each frame is captured over a 1/30 second duration. Each of the bright spots in the image represents a cone, and the dark streaks are shadows cast by blood vessels. The slow vertical scan of the AOSLO causes each row to be captured at a slightly different time. This vertical scan causes the middle frame to be distorted. It appears to have larger cones only because the eye quickly moved vertically during the frame's capture.

The AOSLO is also highly capable for stimulating cone cells as well as imaging them. The AOSLO's scanning of the retina enables simultaneous imaging and stimulation using the same optical pathway. One laser wavelength is used for imaging, and the other is used for stimulation [20, 8]. By modulating the intensity of the imaging and stimulating beams, we can control which photoreceptors are imaged and their degree of stimulation. Since the PMT is tuned to the imaging beam's wavelength, we can use both the imaging and stimulation beams simultaneously. For imaging, we use an infrared 840 nm wavelength beam to illuminate the retina. We choose 840 nm to minimize cone responses to the imaging beam. For stimulation, we use a visible 543 nm wavelength beam. By modulating the intensity of this beam during the AOSLO's scan of the retina, we project a visible image onto the retina similar to a CRT.

For this project, we made use of the AOLSO at Roorda Lab[5] located in UC Berkeley.

[5]http://roorda.vision.berkeley.edu/

Chapter 2

Wizard: The Software for Oz Vision

The AOSLO is the hardware which makes Oz Vision possible, and Wizard is the software control layer which aims to make Oz Vision a reality. Wizard's goal is to be a platform for drawing arbitrary graphics with photoreceptors as the fundamental display element, rather than pixels. Although an ordinary display has direct access to the intensities of its pixels, our hardware and software must circumvent the eye's optics and motion in order to access the photoreceptors. The hardware deals with the eye's optics, and provides Wizard with an image of the photoreceptors and the means to send microdoses to these photoreceptors. Wizard itself has many technical requirements built on top of this input/output in order to create the Oz Vision display.

2.1 Design Objectives

Wizard's final design depends on satisfying many complex objectives simultaneously. These objectives are a combination of constraints on how Wizard must operate given the challenges it faces, as well as important considerations on how to make Wizard a robust technology usable well into the future. The most prominent of these objectives are:

- **Low Latency and High Throughput**: Wizard's control over the AOSLO's stimulation beam must react to eye movements in only a few milliseconds. These reactions must happen at near-kilohertz rates.
- **Programmable**: Third-party programs running alongside Wizard should be able to send commands to update Wizard's graphical output much like an ordinary display. These commands should be as high level as possible to facilitate quick development on top of the Wizard platform.
- **Profiled**: Wizard needs to generate log data detailing exactly what was sent out on the AOSLO and why. This is for use in both debugging and in analyzing experiment data collected using Wizard.

- **Extensible**: The codebase needs to be flexible enough to enable rapid addition of new features, such as support for new hardware. A well-maintained Wizard will also enable contribution from other programmers.

The rationale for each of these objectives and their implications for Wizard's implementation are detailed in this section.

Low Latency and High Throughput

Wizard must track the eye's movement at high precision in order to deliver its microdoses on-target. This precision must be high enough to localize individual cone cells. Since the size of cone cells increases with distance (called *eccentricity*) from the fovea, our tolerance for misdeliveries depends on where on the retina we want to stimulate. At the fovea (0 degrees eccentricity), cones are packed at approximately 15,000 per square degree [23, 5], giving a diameter of approximately 0.5 arcminutes. A microdose that targets the center of a foveal cone must minimally be within 0.25 arcminutes, otherwise we will be stimulating a neighbor cell more than our target cell. In this work, we choose to stimulate at approximately 1.5 degrees eccentricity, with a packing of about 5,500 per square degree [12]. In our test subject, cones are roughly 1 arcminute in diameter in this region. This gives a higher tolerance of around 0.5 arcminutes for a successful delivery. We will introduce more sophisticated methods of error quantification in Section 2.2, but this simple approach is sufficient for discussion.

Wizard's eye tracking must also operate at high speed and low latency, because the eye is under constant involuntary motion. These eye motions occur even when fixating at a stationary location, and consist of microsaccades, tremor, and drift [10, 14]. Microsaccades are quick jumps from one location to another, and between these microsaccades are periods of eye drift. The average microsaccade occurs in less than 30 ms, and travels at speeds of over 1200 arcminutes per second [14]. The combination of tremor and drift cause movements similar to Brownian motion, with frequent direction changes at significant velocity [17]. An example of eye drift is shown in Figure 2.1. We cannot accurately predict these eye drifts, and so our strategy is to send microdoses relative to the last known location. This is because eye drift tends to hover around zero net movement, even if each individual path is largely unpredictable.

The random-walk nature of eye drift implies that latency is the limiting factor in how accurately we can deliver microdoses. If there is a Δt ms delay between when we measure the eye's location and when we update the microdose targets, then we will miss by however much the eye moved during the Δt ms. If our tolerance is 0.5 arcminutes, then we need to decrease Δt until the eye does not move more than 0.5 arcminutes within that time. Figure 2.2 shows how decreasing the Δt latency affects our probability of a successful delivery.

Based on Figure 2.2, if we can achieve under a 4 ms latency, we can deliver microdoses with over 90% within a cone's radius at 1.5 degrees eccentricity. When considering that the

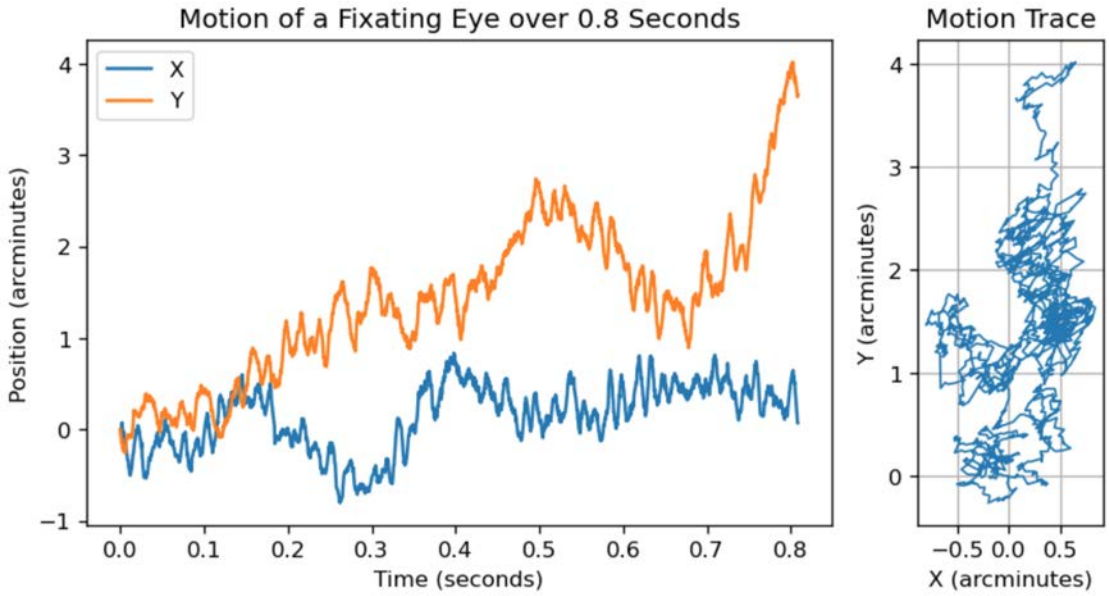

Figure 2.1: An example of eye drift motion over a 0.8 second interval. Left: The X and Y position of the retina as a function of time. Right: The same motion trace shown as a walk in the XY plane. Origin is set at the starting position. This eye drift is bounded by two microsaccades not shown. Measured on the AOSLO.

eye is moving like this constantly, the immediate consequence is that Wizard also needs to perform this calculation *every* 4 ms, or at least at 250 Hz.[1]

Programmable

Wizard is intended to be a general-purpose display tool for per-photoreceptor graphics, and so it must provide an interface for outside programs to control these graphics. We call this making Wizard "programmable," because it allows the end user to build extra application logic on top of the controls Wizard exposes.

These controls are not geared to a singular application, but our area of focus is to enable *psychophysics* experiments to be built with them. Such experiments measure how a test subject internally experiences a given stimulus. These experiments are critical for characterizing the colors perceived with our display, as well as for validating the models that predict those perceptions.

Wizard must be compatible with established scientific procedures for collecting psychophysics data on a computer display. Such scientific procedures often feature special-

[1]These numbers are generous bounds on how slow Wizard should be allowed to run. In practice we will also deal with imperfect tracking and saccades. Our current implementation runs at 2-3 ms of latency at 1,000 Hz through parallelism. The empirically measured performance is shown in section 4.1.

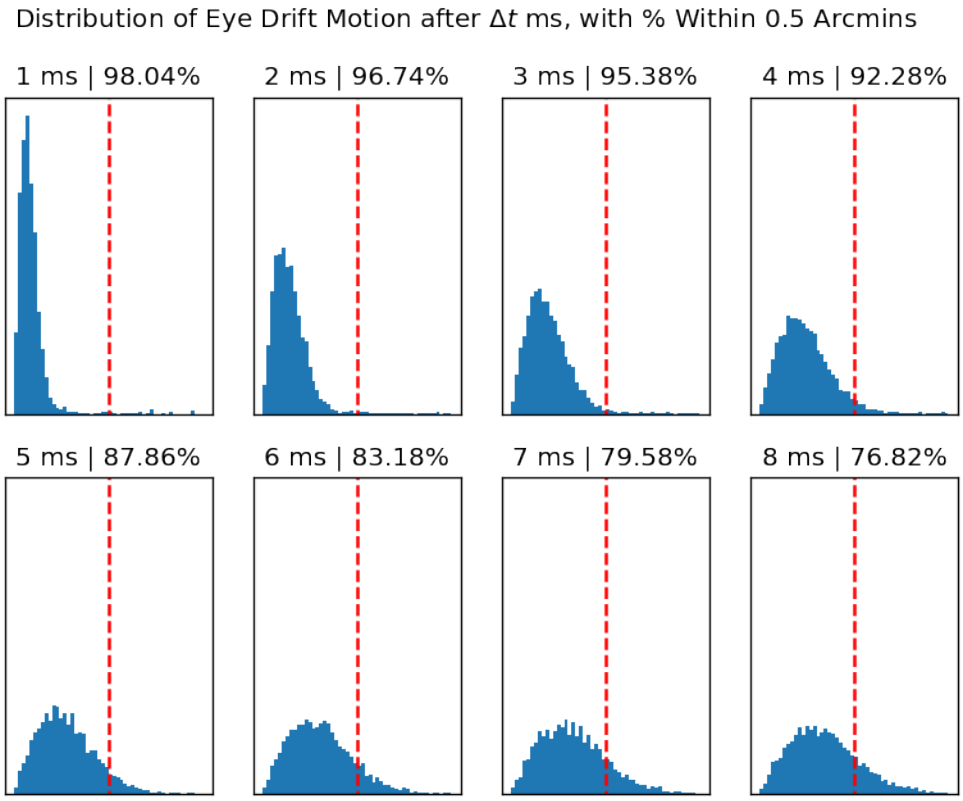

Figure 2.2: Histograms showing the total distance traveled by the eye over a Δt time interval. These statistics were collected over the same motion plotted in Figure 2.1. The horizontal axis is 1 arcminute wide, with the dashed red line showing our tolerance of 0.5 arcminutes. Each subplot is titled with the time in milliseconds, and the success rate, which is the percent of displacements within 0.5 arcminutes. At around 4 ms, we achieve over 90% success.

ized computer scripts for precisely controlling what is presented to the test subject, and for recording their responses. In this context, Wizard is just another display device, and so needs to expose high-level controls for these external scripts. A graphical user interface alone is not enough, because this only offers a manual level of control.

For example, Wizard exposes a control to draw a rectangle of a specified size uniformly filled with a given LMS tristimulus value. This simple control could be used to design an experiment to determine what color a test subject perceives from the given LMS tristimulus value. An external script controlling both Wizard and a traditional display would show the LMS tristimulus value alongside a known colored rectangle of the same size and energy. This script could then accept keyboard input from the test subject to adjust the traditional display's color until it matches Wizard's, or to specify that no match could be made. By repeating these presentations with different LMS values, we get many data points mapping

from Wizard LMS values to colors shown on the traditional display. This rich behavior was made possible while requiring only minimal development effort on Wizard's part.

Profiled

Wizard must keep a comprehensive record of its inputs, outputs, and computation during a live experiment. This is needed to quantify errors in delivery, which are any discrepancies between what Wizard intended to deliver to the retina versus what actually arrived. These errors can arise from latency as described earlier, but can also result from simple software bugs. This capability is especially relevant early in Wizard's development, where Wizard may not be fully tuned but is still able to power psychophysics experiments. A record of the inputs received from both the hardware and the user is needed to explain how Wizard ended up sending the outputs that it did.

Wizard's log files should minimally contain:

- The sensory data from the hardware. In the case of the AOSLO, this is the image of the retina which is used to locate the cones.
- The intended activity levels for each cone cell, at each point in time. This is a record of what the user requested from Wizard.
- The microdoses sent to the eye through the hardware, computed by Wizard in an attempt to achieve the intended activity levels. In the case of the AOSLO, this is the modulation of the stimulation beam.

The input data is needed to know where the cones actually were relative to the microdoses that were sent. This will enable us to compute the actual activity levels and compare these to the intended activity levels. We go into more detail on this in Section 2.2.

The ability to analyze Wizard's performance after the experiment is over can also offset the deficiencies in an in-development Wizard. Building off of the color-match example given earlier, we can condition each color match on whether or not Wizard actually delivered what we intended to. Even if Wizard has a 50% chance of failure for each presentation, we can simply record twice as many data points and separate out the failures after-the-fact.

The specialized data contained within these Wizard log files will appropriately need specialized tools for analyzing them. These tools should provide visualizations of the microdose deliveries, as well as compute error metrics to qualitatively measure Wizard's performance. We go into more detail on these tools in Section 2.2.

Extensible

We cannot anticipate every possible use-case for Wizard, and so we cannot predict how Wizard may evolve through future iterations. The design objectives outlined above are relevant for any future Oz Vision display, but we may discover new requirements or applications for

Oz Vision during the development process. Therefore, we also want Wizard's codebase to be easily extensible to include new features as we discover the need for them.

For example, although Wizard currently runs on an AOSLO, it should not be written to work *only* on an AOSLO. There may be future imaging and stimulation hardware which we also want to use for Oz Vision. In that case, we do not want to repeat development of another Wizard-like software suite on that new platform. Wizard should be flexible enough to just add in support for this new platform, allowing re-use of its core Oz Vision features.

2.2 Post-Experiment Analysis with Wizard Log Files

Wizard's log files contain a detailed record of what stimulation levels were intended and what microdoses were sent in an attempt to achieve those intended levels. By analyzing these log files, we can determine to what extent the microdoses achieved the intended stimulation levels. The precision of the log files lets us compute different metrics for measuring how successful a delivery was.

There are two key error metrics: **Motion Prediction** and **Stimulation** error. We describe these in greater detail below, and Figure 2.3 gives a high-level overview.

Motion Prediction Error

Motion prediction error is the difference between the spatial (x, y) locations we *intended* the microdoses to land versus the locations that they actually *arrived* at.

Due to latency between when the eye moves and when Wizard is able to sense this movement, Wizard always works with a slightly outdated picture of where the cone cells are. Therefore, Wizard needs to predict where the eye currently is, using only data from the past. This is why we call this error "motion prediction" error. Minimizing the motion prediction error can either be done by decreasing latency, or improving the accuracy of the prediction.

Stimulation Error

Stimulation error is the difference between the activity levels we *intended* to achieve with the microdoses versus the activity levels that *arrived* from those microdoses.

For example, although $\texttt{LMS}\begin{bmatrix}0 & 1 & 0\end{bmatrix}$ might be our intent, $\texttt{LMS}\begin{bmatrix}0.1 & 0.9 & 0\end{bmatrix}$ might be what ends up arriving if some microdoses intended for M cells end up hitting L cells by mistake. Possible causes include high motion prediction error, and light being improperly focused.

Getting the *arrived* LMS values from the log files requires a separate model for how to translate a distribution of photons into an LMS activity value. One such model would simply implement the details of human vision given in Sections 1.1 and 1.2.

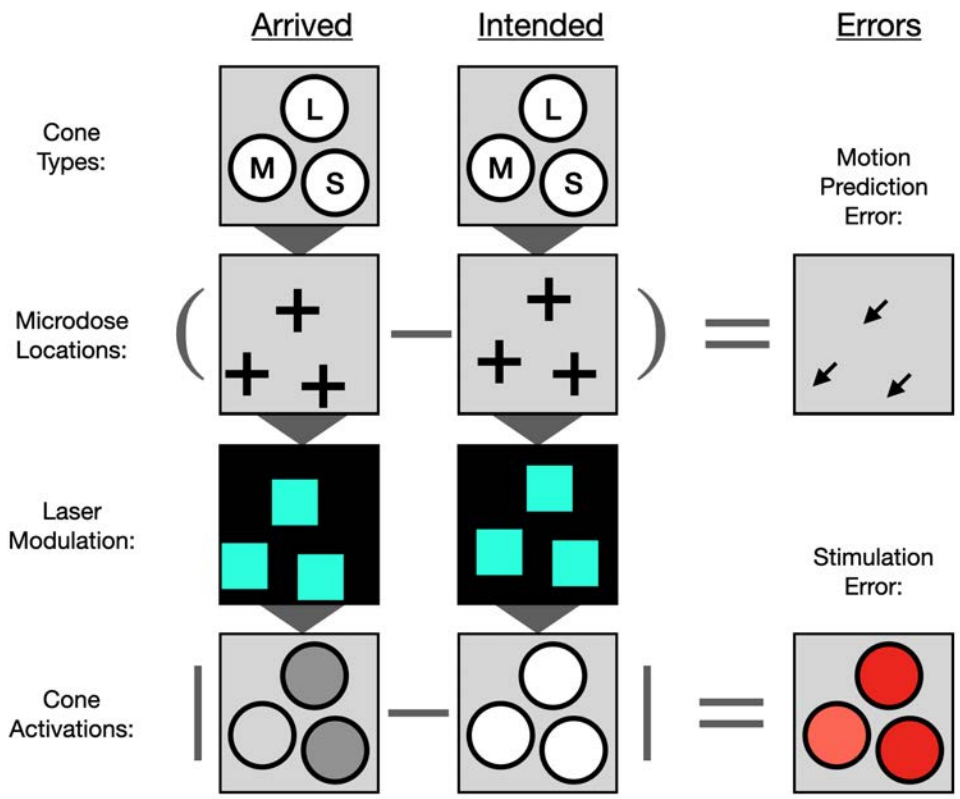

Figure 2.3: Motion prediction and stimulation error are differences between what arrived at the retina versus what Wizard intended to deliver. In this example, the microdoses arrived down and to the left of their targets. The left column is what arrived at the retina and the middle column is what Wizard intended to be received. Both columns show the top-down progression from Wizard input to Wizard output, and are derived from the log files. The right column shows the two error types as a difference between the delivered and intended data at different stages in the pipeline. Motion prediction error is the vector between the microdose locations. Stimulation error is the absolute difference between the activity levels.

More generally, we can compare the intended and arrived stimulation levels on a cone-by-cone basis. If i is the intended stimulation level, and a is the arrived stimulation level, then we can write the "stimulation error" as $|i - a|$.

Compared to motion prediction error, stimulation error is a better metric when determining if Wizard's deliveries were successful or not. For example, high motion prediction error will matter less if we intended `LMS` $\begin{bmatrix}1 & 1 & 0\end{bmatrix}$ instead. Even if we hit an L cone when we intended to hit an M, this mistake is less important since we wanted to hit L cones anyway.

Lastly, using psychometric data, we can use stimulation error not just as a measure for Wizard's performance, but as a measure for the correctness of the model used to compute

stimulation error. Ideally, we can use stimulation error to make a prediction about what color a test subject saw instead of what we intended them to see. By comparing a test subject's perception of the color against what we predicted them to see, we can refine that predictive model. A similar method is later used in Chapter 5.

Chapter 3

Wizard's Software Engineering

This chapter goes into detail on the specific software engineering choices made to satisfy the technical requirements outlined in the previous chapter.

3.1 Software Flexibility Through Modularity

Wizard is written following modern software practices in order to make its code structure amenable to the features and technical requirement shifts we will discover on the road to Oz Vision. We chose to write Wizard in C++, an object-oriented language for writing performance-sensitive applications while still enabling flexible software architectures.

The first step in implementation was to divide up Wizard's goal of Oz Vision into smaller self-contained modules. Each module consumes data provided by earlier modules, while also producing data for later modules to use. Each module is self-contained in the sense that it only cares about getting the data that it needs, but does not care about how it is produced. Along with this idea, each module should strive to produce data which is as broadly useful as possible. This idea is also known as the observer pattern [7], and is widely used in software engineering to prevent code from becoming too specialized to a singular end goal.

There are four main modules that Wizard is composed of:

- **Sensing**: This module is responsible for receiving input data from the optical hardware. Its purpose is to simply echo this input data to the downstream modules, along with other useful metadata such as the arrival timestamp.
- **Tracking**: This module receives sensory data from the sensing module and uses this to perform eye tracking. This eye tracking data is a time-varying description of how the eye is oriented relative to the optics hardware.
- **Rendering**: This module is where we decide what time-varying activity levels we want the cone cells to be at. There is no restriction on what kinds of data this module may need, since there are limitless possibilities for what an end user may want to display on the retina.

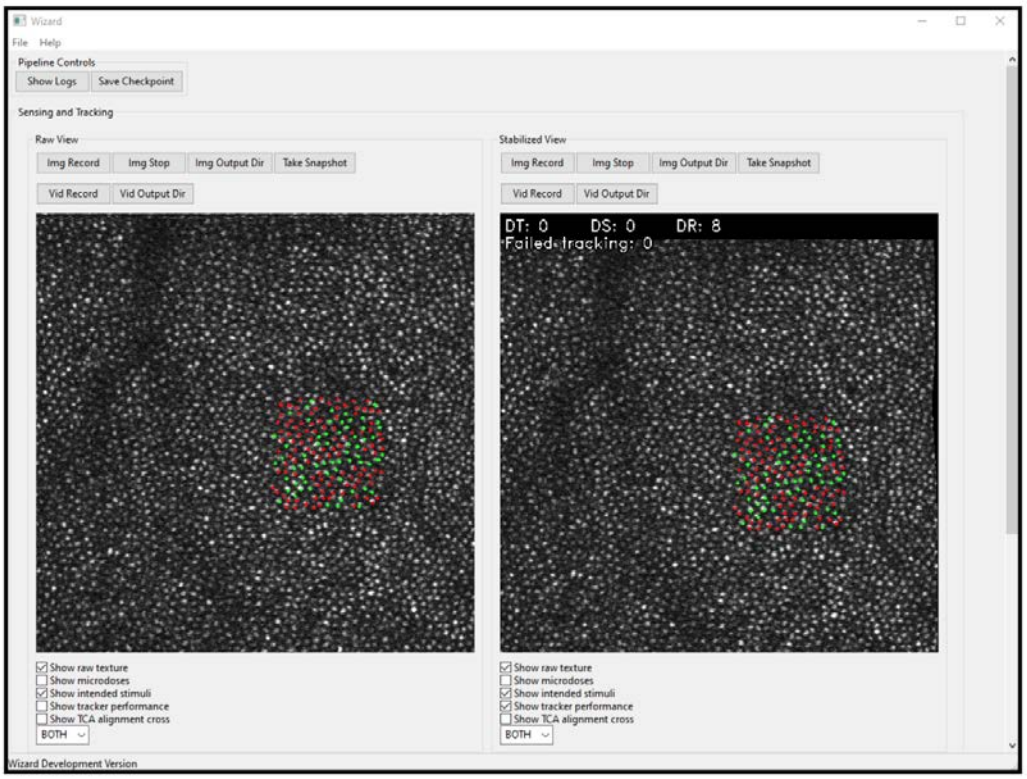

Figure 3.1: The Wizard graphical user interface (GUI). Shown are two views of the retina during a single presentation of `LMS` $\begin{bmatrix}1 & 1 & 0\end{bmatrix}$ across a 0.25×0.25 degree rectangle. Left: ("world-fixed") the view of the retina as seen by the AOSLO. Right: ("retina-fixed") the same image except warped such that the cones are fixed in place within the GUI. Both views have an overlay of the L/M intended activity levels being delivered shown in false color (red/green is L/M). The significance of having two views of the same data is illustrated in Figure 3.2.

- **Rasterization**: This module receives the list of intended cone activity levels, and will send commands to the optics hardware in order to best achieve those levels. Naturally, this module requires data from the tracker so that it can determine where to send microdoses.

Splitting up Wizard into self-contained modules has significant performance benefits by running each module in parallel. For example, if module B uses outputs from module A, then B can process the first output at the same time that A is preparing to send the second output. This method is also known as "pipelining," and is at the core of how Wizard maintains high end-to-end data throughput. However, the time required to go all the way from sensing to rasterization does not change, since each module individually still takes the same amount of

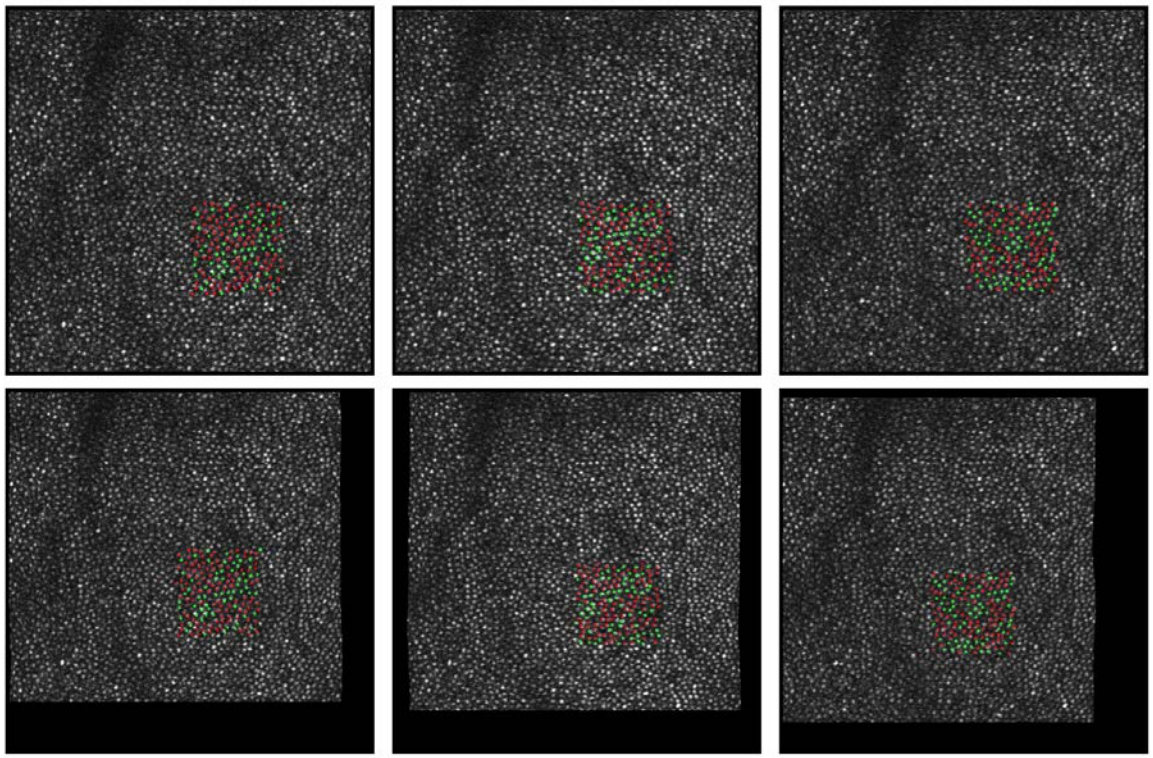

Figure 3.2: Comparison of the world-fixed and retina-fixed views during a single presentation of `LMS` $\begin{bmatrix}1 & 1 & 0\end{bmatrix}$ across a 0.25 $\times$ 0.25 degree rectangle. Top row: three frames during the same presentation, shown in world-fixed view. Bottom row: the same three frames, shown in retina-fixed view. The world-fixed view shows the motion of the eye as the movement of the gray retina texture as seen by the AOSLO. The retina-fixed view is the same frame data but with the motion negated so that the cones stay fixed in place. The black boundary indicates that those regions of the retina were not imaged in that frame. Both views have an overlay of the L/M intended activity levels being delivered shown in false color (red/green is L/M).

time to convert inputs into outputs.

This software pattern also makes adding logging and a graphical user interface (GUI) much easier. Each module does not need to be aware of how its data is being used, and so one such use is to just save the outputs to disk. This is what enables Wizard logs to be a comprehensive record of how it operated during a live experiment. Additionally, since modules do not need to know where its inputs come from, a GUI was written to just broadcast data updates to these modules. This GUI also functions similar to the logger, except the outputs are displayed graphically to the user. This GUI is shown in Figure 3.1.

3.2 Fast and Robust Tracking

Wizard uses the AOSLO's live video of the retina to track the eye's motion. This method of AOSLO-based eye tracking was first described by Stevenson *et al.* [21], and our approach is a modification of their method. AOSLO-based eye tracking provides for the spatial and temporal resolution we need for Wizard's latency and speed objectives.

The AOSLO features a slow vertical sweep across the retina at 30 Hz, which creates a rolling-shutter video at 30 fps. Each row of a frame is captured at different time intervals, sequentially organized from top to bottom. Therefore, the bottom row is captured 1/30 of a second after the top row. This rolling-shutter causes distortions in the image if the eye moves, shown in Figures 1.7 and 3.3.

Incoming frame data is registered against a *retina map*, which is a pre-existing image of the retina under no motion. Since the AOSLO is fixed in the world, any motion in the live video is due to motion of the eye. By registering these distorted frames against the distortion-free map, we can recover the distortions in the frames. These rolling-shutter distortions encode the motion of the eye, allowing us to do eye tracking.

If we were to wait for an entire frame to arrive before doing registration, then we could never reach the desired 250 Hz motion tracking rates. Therefore, we perform partial registrations on horizontal "strips" of frame data as it arrives in Wizard's memory. Each 512 × 512 incoming frame is split into 32 strips each 512 × 16 in size, representing approximately 1 ms of capture time. Each strip registration then provides a single estimate of the eye's position at the time of its capture. By registering each strip in real time, we achieve a tracking sampling rate of 1 kHz. This strip-by-strip registration is shown in Figure 3.3.

For image registration, we use normalized cross-correlation (NCC) [11]. NCC finds an (x, y) position for each strip in the retina map. This (x, y) position maximizes the correlation coefficient between a strip and the overlap it has with the retina map. Let $M_{x,y}(u, v)$ be the 2D region of the retina map overlapping with the strip $S(u, v)$ placed at position (x, y). Let $\hat{M}_{x,y}$ and $\hat{S}$ be their respective means. Then the normalized cross correlation is:

$$\gamma(x,y) = \frac{\sum_{u,v}\left(S(u,v)-\hat{S}\right)\left(M_{x,y}(u,v)-\hat{M}_{x,y}\right)}{\sqrt{\sum_{u,v}\left(S(u,v)-\hat{S}\right)^2\sum_{u,v}\left(M_{x,y}(u,v)-\hat{M}_{x,y}\right)^2}}$$

The output of NCC is the choice of (x, y) which maximizes γ. NCC is a rigid registration, which causes all of the motion within a single strip to be averaged out into a single (x, y) value. In principle, we could use a non-rigid method, but we found that NCC works well in practice. Most importantly, we were able to speed up NCC to run at 0.33 ms per strip. This was done by implementing fast normalized cross correlation [11] on the GPU. Our implementation uses hand-optimized CUDA code[1] running on an NVIDIA GPU[2] [15]. However, we cannot

[1]Thanks to Yi Zong and Dr. Emma Alexander for the prototype implementation. The final version presented here is a complete re-write.

[2]We used an NVIDIA Titan V, although our code definitely does not make full use of it.

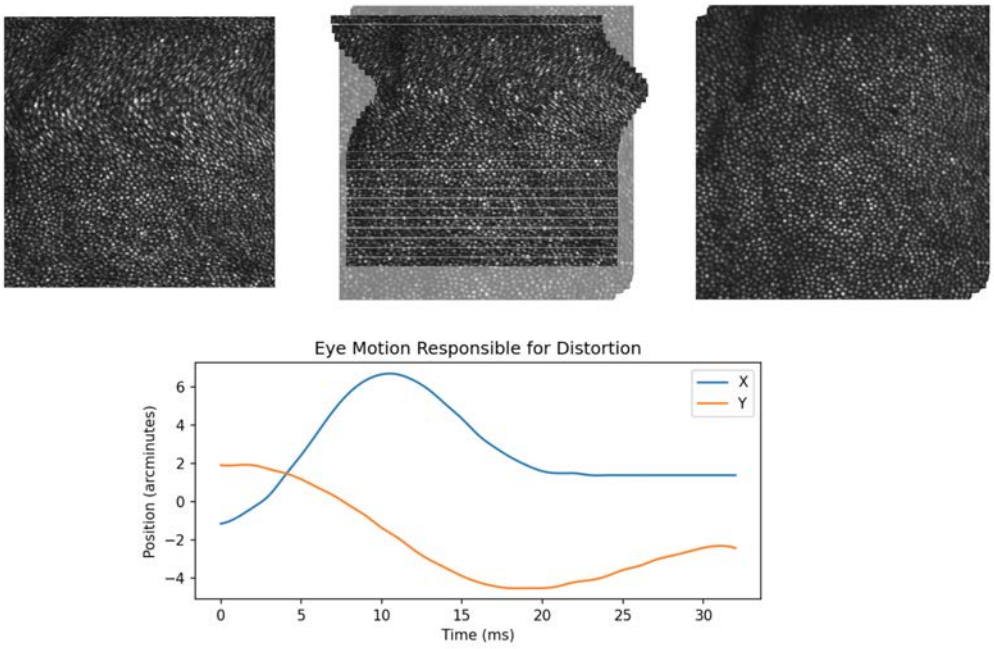

Figure 3.3: Strip-by-strip registration an AOSLO frame against a pre-existing retina map. Top Left: The raw frame, as it arrives from the AOSLO. Top Right: The retina map, acquired beforehand. Top Middle: The raw frame registered onto the retina map on a strip-by-strip basis. Bottom: The movement of the retina which caused the rolling shutter distortion in the raw frame. The incoming frame is split into 1 ms intervals, which correspond to 512 $\times$ 16 strips since the AOSLO captures each row sequentially in time. This enables us to record the eye's motion at high temporal and spatial resolution. Each strip registration gives a single estimate for how the retina is translated relative to the map.

use NCC on its own to get a confidence[3] for the returned (x, y) position. Therefore, using the NCC outputs at face value is error-prone due to false matches. This is especially true during microsaccades, which cause extreme rolling shutter distortions that are challenging for this rigid registration method. An example of these false matches during eye drift are shown in Figure 3.4.

To eliminate false matches, we use RANSAC on a per-strip basis [6]. For each strip, we divide it into n substrips with vertical slices. Then, each substrip is individually registered onto the retina map through NCC. Each substrip is therefore its own sample of the eye's position. We run RANSAC on these samples in order to filter out the false matches. Since we are solving for eye position, the model is determined by just a single sample. The consensus threshold is then for a majority of the remaining $n-1$ samples to be within a given radius of

[3]We attempted to threshold on the maximum γ which NCC finds, but there is no consistently good value for this threshold.

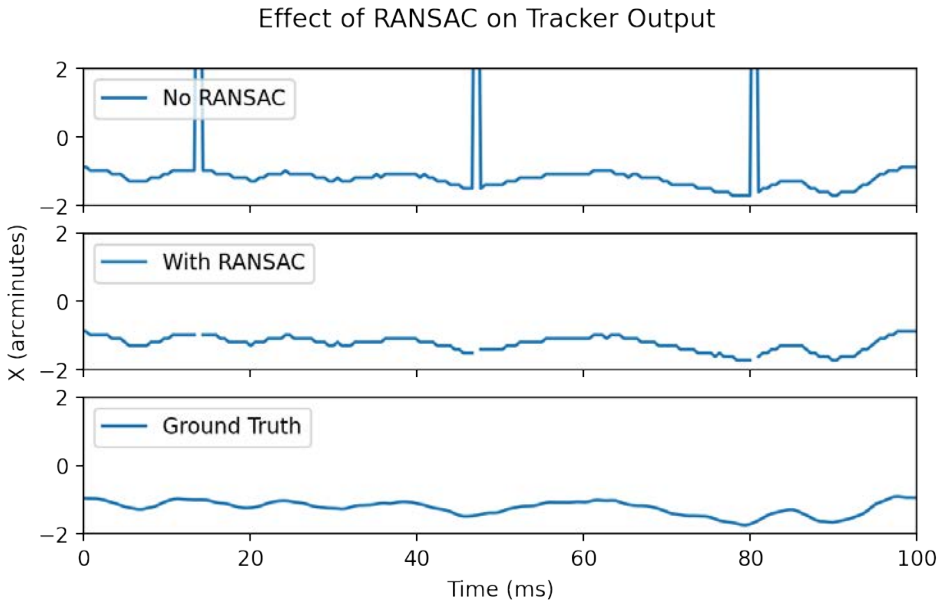

Figure 3.4: Eliminating false NCC matches using RANSAC. We extracted the motion from a 100 ms video of eye drift. Top: Tracker using raw NCC only. Middle: Tracker using NCC with RANSAC to remove outliers. Bottom: Ground truth. The vertical spikes in the raw NCC trace are false matches. These artifacts are removed with RANSAC, shown by the gaps where the tracker reported that it could not find the eye's position.

that single sample. We found that choosing a radius of 0.5 arcminutes and $n = 3$ works well in practice. Once a consensus is found, we output the average of the non-outlier samples as the final eye position for that strip. If no consensus can be found, then the tracker reports failure instead. The effect of using RANSAC is shown in Figure 3.4.

3.3 Matlab Programming Interface

Wizard allows external programs to control it via inter-process communication (IPC) to give users the power to integrate Wizard into their existing software infrastructure. Many useful features, like updating the display in response to a test subject's keyboard press, do not actually depend on the complex code for Oz Vision. Rather than bring these features into the scope of Wizard's development, we can instead allow external programs to implement these features on top of Wizard. This frees up Wizard development to focus on the core challenges facing Oz Vision, while enabling third-party developers to build extra features on top of Wizard without interfering with core development.

To maximize the breadth of features that can be built on top of Wizard, we expose a set

of Matlab functions which send IPC commands to Wizard. Matlab is a scripting language and development environment which is already widely adopted by the scientific community, and so this Matlab interface caters specifically to this audience. The only limit to controlling Wizard is our choice of what controls to expose.

An example Matlab program built on top of Wizard is given in Figure 3.5, where we display a 0.5×0.5 degree rectangle uniformly filled with `LMS` $\begin{bmatrix}1 & 2 & 3\end{bmatrix}$.

```
% Rectangle boundary, in degrees.
min_x = 0.25; max_x = 0.75;
min_y = 0.25; max_y = 0.75;

% The desired LMS tristimulus value.
l = 1.0; m = 2.0; s = 3.0;

% The number of frames to display for.
num_frames = 30;

% Tell Wizard that we want a world-fixed rectangle
% with the given LMS tristimulus value and boundary.
SetRenderMode('world-fixed');
SetWorldFixedLMSRectangleParams(l, m, s, min_x, max_x, min_y, max_y);

% Display the stimulus for our desired duration.
DisplayFor(num_frames);
```

Figure 3.5: Example Matlab code for displaying a 0.5×0.5 degree rectangle uniformly filled with `LMS` $\begin{bmatrix}1 & 2 & 3\end{bmatrix}$. The "world-fixed" mode refers to how the rectangle should appear to the test subject as though it is not moving relative to the AOSLO.

3.4 Error Analysis and Visualization Toolkit

Since Wizard needs to run at high speeds, the log files it produces are what is minimally necessary to enable thorough offline investigation into its online behavior. Therefore, specialized tools are needed to extract interpretable information from these dense log files.

For every strip, Wizard saves a log of:

- The image of the retina received from the AOSLO.

- The tracker's estimate of the retina's (x, y) position, and how outdated this estimate was at the time of delivery.
- How Wizard modulated the stimulation beam, saved as:
  - A list of all the cone cells that were targeted for delivery during this strip.
  - Said cone cells' intended activity levels.
  - A list of the microdose locations and intensities used to achieve those activity levels.
- Timestamps for when all of the above events occurred.

Alongside the per-strip data is a record of the retina map used, which is needed for computing a more accurate eye motion trace from the logged images of the retina.

The Error Analysis and Visualization Toolkit (EAVT) is a collection of offline tools developed alongside[4] Wizard for processing these dense log files. The log files only implicitly provide the motion prediction and stimulation errors, and so the EAVT spends compute time to make them readily available. The EAVT is also equipped for visualizing both these errors and the raw data itself for further analysis.

A core part of the EAVT is its model for deriving the arrived activity levels from the intended microdoses. This is needed because the logs only specify what was sent to the eye, and not the activity levels that those deliveries elicited. A visualization of this model in action is shown in Figure 3.6 for a `LMS` $\begin{bmatrix}1 & 0.25 & 0\end{bmatrix}$ intended stimulus spanning a 0.25×0.25 degree rectangle.

Using the EAVT, we can visualize and measure the motion prediction and stimulation errors as described earlier. These visualizations are shown in Figures 3.7 and 3.8 respectively. Due to the AOSLO's slow vertical scan, the motion prediction errors on the same rows are identical, since they correspond to nearly the same point in time. The errors shown here correspond to a latency of approximately 3 ms. Closely related are the stimulation errors, which take into account the actual mechanics of the eye. Notice how areas with less motion prediction error also have less stimulation error. Given how stimulation error depends on a complex and possibly inaccurate model of the eye, we often just compute motion prediction error when tuning Wizard performance. However, stimulation error is ultimately the metric that captures the nuances of an Oz Vision presentation.

[4] Although the EAVT is in some sense an extension of Wizard and shares significant amounts of code, for clarity we refer to the EAVT as though it is a separate program. The EAVT sub-project was originally started by Varsha Ramakrishnan during Summer 2020, and James Fong continued development starting Fall 2020.

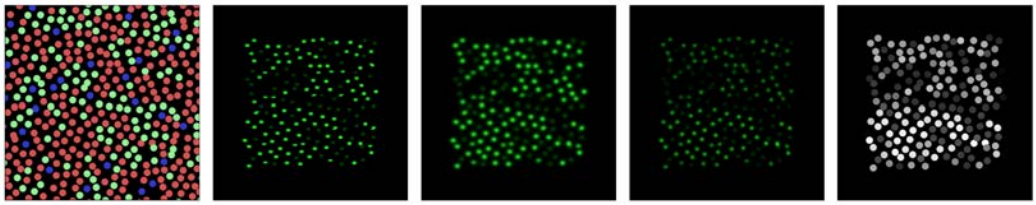

Figure 3.6: The EAVT's model for computing arrived cone activities from the logged intended microdoses and the retina map. Each view shows a 10 × 10 arcminute section of the retina. From left to right: 1) The map of the retina, with cone identities shown in false color (red/green/blue is L/M/S). 2) The arrived microdoses. 3) The arrived microdoses after optical blur. 4) The microdoses viewed through each cone's aperture, similar to column 4 in Figure 1.3. 5) The computed arrived activity levels after spatial and spectral integration.

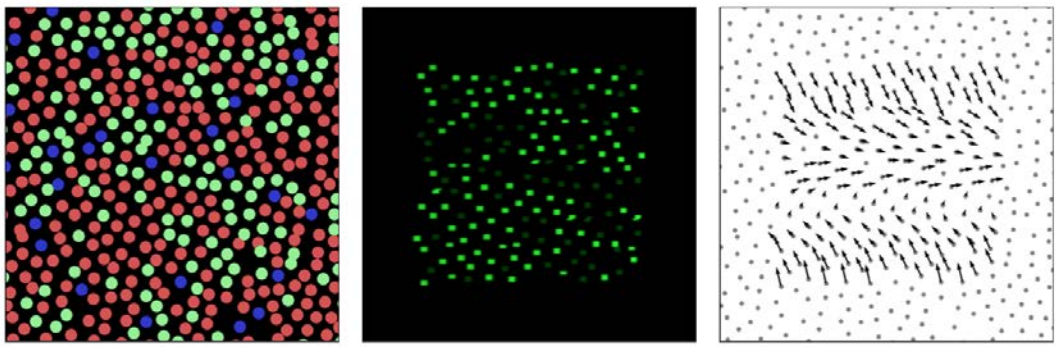

Figure 3.7: Motion prediction error visualized with the EAVT. Each view shows a 10 × 10 arcminute section of the retina. From left to right: 1) The map of the retina, with cone identities shown in false color (red/green/blue is L/M/S). 2) The arrived microdoses. 3) The difference between each cone's location (equivalently, the intended microdose location) and its corresponding arrived microdose's location. The error vectors are scaled 4x to show detail.

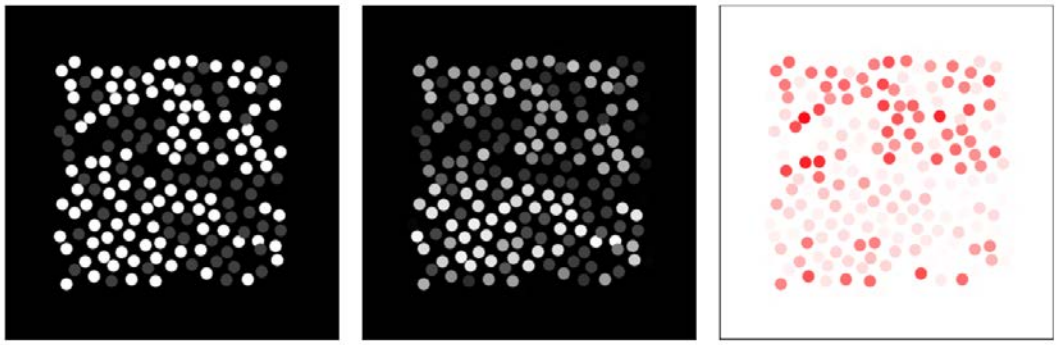

Figure 3.8: Stimulation error visualized with the EAVT. Each view shows a 10 × 10 arcminute section of the retina. From left to right: 1) The logged intended activity levels. 2) The computed arrived activity levels. 3) The absolute difference between the intended and arrived activity levels, with a more intense red meaning a greater difference.

Chapter 4

Validating Wizard's Performance

The complexity of Wizard demands a thorough validation before it can be confidently used to collect scientific data. Strictly speaking, a human test subject is only required to gather psychometric measurements of the stimuli presented. All other debugging efforts ought to be done through other methods, since including a human test subject introduces substantial difficulties. For most debugging on the AOSLO, we only need the image and motion statistics of a real human retina. In this chapter we detail the methods used to validate Wizard as much as possible without requiring human test subjects. Each method has its benefits and trade-offs, but are nonetheless essential for thoroughly testing Wizard components in isolation.

4.1 Validating Tracking Accuracy in Replay Testing

The simplest method for testing Wizard without requiring a live human test subject is to just run Wizard on a pre-recorded video of a live retina. Owing to Wizard's flexibility, the module responsible for interfacing with the AOSLO can be swapped out with a module which replays real AOSLO inputs. We call this technique "replay testing."

Replay testing has the added benefit of testing Wizard under different hypothetical settings. One such application is testing the effect of latency on the tracking performance. This is done by running Wizard's tracker on a pre-recorded video multiple times at different simulated latencies. We then can compute the prediction error against a high-quality ground truth for the source video. The results are shown in Figure 4.1.

When combined with knowledge of Wizard's real time performance, this data enables us to make predictions on the robustness of Wizard. In the case of tracking, although Wizard runs with 2-3 ms of latency during live sessions, Figure 4.1 suggests that we could go as high as 4 ms while still landing our stimuli within a cone's radius 75% of the time.[1] This value

[1]This is much lower than the 90% value given in Section 2.1. This is because this measurement takes into account high-velocity saccades and tracker inaccuracies, whereas the earlier estimate uses only a ground truth eye trace during eye drift.

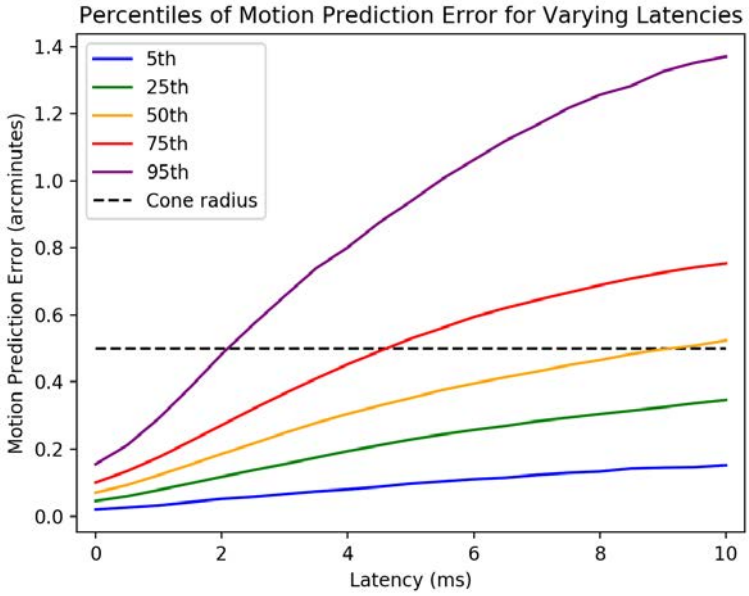

Figure 4.1: Prediction error for Wizard running at different simulated latencies via replay testing. For a given predicted location, we compute the Euclidean distance against a ground truth. Shown are the percentiles for different error levels. The dashed line indicates approximately the radius of a cone cell at 1.5 degrees eccentricity.

can still be improved, but at the current speeds, Wizard is performant enough to warrant preliminary tests on real human eyes.

4.2 Validating Tracking and Laser Modulation with a Model Eye

Usually testing simulation is sufficient, but often we need to validate Wizard's performance when running on the real AOSLO hardware. Since modifying the hardware would invalidate such a test, we instead simulate the behavior of the eye. This is done with the "model eye," a hardware component placed where a test subject's eye would normally be. This model eye imitates the texture of the retina, and its movement can be programmed to imitate the velocities found during real eye drift. The image and motion quality of the model eye is shown in Figure 4.2 and Figure 4.3.

Using this model eye, we can test Wizard's accuracy when delivering microdoses to features on the retina. To do this, we use a trick to encode the laser pulse locations directly into the raw image recorded by the AOSLO. Rather than increase the intensity of the stimu-

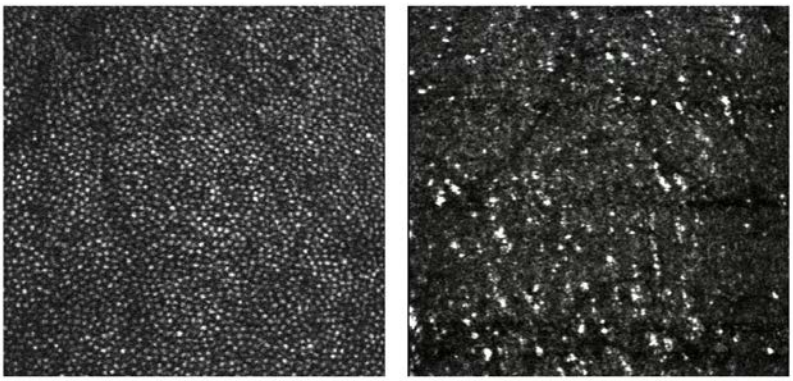

Figure 4.2: Comparison of the texture of a real eye's retina (left) and that of the model eye's retina (right). Although the textures are distinct, the purpose of this texture is to provide Wizard's tracker with a suitable image to track against.

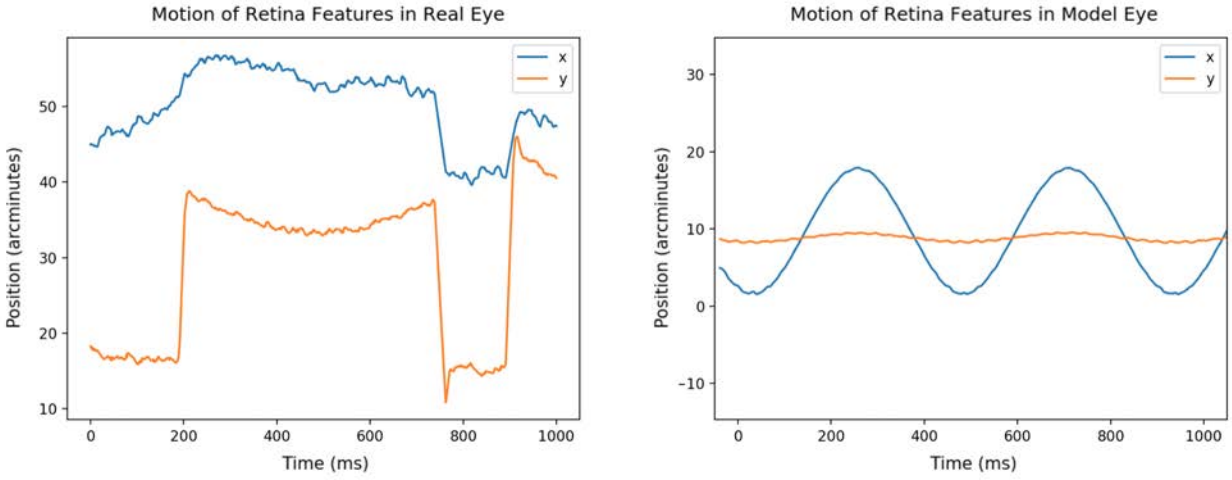

Figure 4.3: Comparison of the motion of a real eye's retina (left) and that of the model eye's retina (right). The model eye lacks the sudden jumps in microsaccades. However, the amplitude of the model eye's motion is adjusted such that the average velocity is similar to that found in real eye drift.

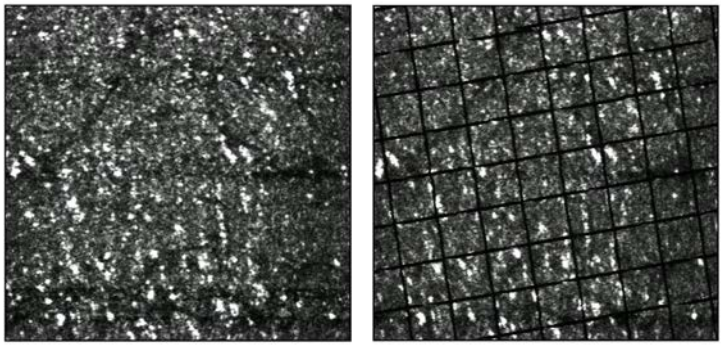

Figure 4.4: A decrement pattern sent on the imaging beam appears in the raw AOSLO image. Here we are sending a grid pattern rotated by 30 degrees.

lation beam at a desired location, we can decrease the intensity of the imaging beam. Such a "decrement pattern" appears in the raw AOSLO image in negative contrast [1]. This effect is shown in Figure 4.4. Notice how the model eye texture is still visible underneath the grid pattern. The black grid lines are not an overlay applied in software; they are directly recorded by the AOSLO since the imaging beam is turned down for those pixels.

We can now jointly test the tracking and modulation accuracy by adding motion to the model eye. Wizard will attempt to keep the decrement pattern fixed relative to features on the retina. In order to do this, it will need to track the retina and move the decrement pattern in the same direction to counteract the motion. This also tests that Wizard is able to update the laser output in real time. If either the tracker or the modulation fail, then the pattern will move relative to the retina. Such translation errors will exactly mimic those that would have occurred when delivering stimulation laser pulses.

The results of this test are shown in Figure 4.5. As the retina moves, the decrement pattern is correctly updated in real time to stay at a fixed location.

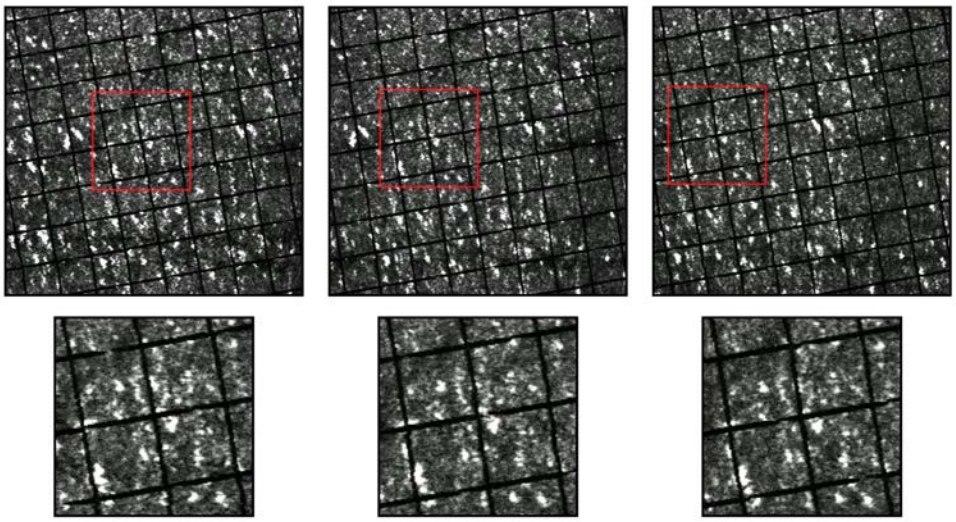

Figure 4.5: A decrement pattern tracking the motion of the retina in real time. Three frames at different points in the model eye's motion are shown. Top: The entire frames, with a red border highlighting an area of the retina. Bottom: The highlighted area, scaled up to showcase the tracking performance.

Chapter 5

Early Experimental Results

Wizard has been successfully used to produce different perceptions of color from a single wavelength light source via spatial structure alone. Wizard's performance has yet to be tuned enough to display the full Oz gamut, but these early experimental results achieve significant milestones on the road to full Oz Vision. They demonstrate the feasibility of Oz Vision in the near future, as well as validate that all of Wizard's components are working end-to-end.

Here we present two results, intended to showcase Wizard's capabilities at its current stage of development.

- Different color percepts were elicited using only a single wavelength laser.
- Attempting to present a pure M stimulus introduces both spatial structure and a noticeable blue tint to the stimulus.

The sessions to collect this data were conducted by Roorda Lab and Tuten Lab personnel following standard protocol for AOSLO stimulus presentations on human test subjects that was approved by UC Berkeley's Institutional Review Board.

5.1 Color Match Design and Limitations

The goal of this experiment was to compare a subject's perception of different L/M activity levels under two different presentations: Oz Vision and uniform light. The Oz Vision presentation stimulated the L and M cone cells to the desired amount by modulating the intensity of a single 543 nm wavelength laser as it passes over each cone. The uniform light presentation was a simple additive mixture of two laser light sources (543 nm and 680 nm) at a uniform intensity.

The choice of wavelengths ensured that all tristimulus values presented were on the LM plane. 543 nm light activates the L and M roughly equally, and 680 nm gets approximately 1 log unit more L activation than M. For both wavelengths, the S cone activation is negligible.

Since the uniform light was just a mixture of 543 nm and 680 nm, all uniform light presentations were on the LM plane. The Oz light used only the 543 nm laser, and so it could only activate some combination of L and M cells as well. Even if Wizard had attempted to target S cones, the 543 nm light would not have been able to significantly activate them.

To test Wizard's ability to display colors on this LM plane, we displayed tristimulus values of different L/M ratios. We chose 5 points uniformly spaced between two tristimulus values: $\texttt{LMS}\begin{bmatrix}1 & 1 & 0\end{bmatrix}$ and $\texttt{LMS}\begin{bmatrix}1 & 0 & 0\end{bmatrix}$. These LMS values were chosen to roughly match the range of LMS values achievable with the uniform light.

For each Oz light presentation, the test subject used method of adjustment to tweak the appearance of the uniform light until the colors were indistinct. For each trial, the Oz light was displayed, followed by the uniform light. The test subject could repeat the two presentations as often as they wanted. Using a keyboard, the test subject gradually adjusted the powers of the two lasers in the uniform light until a match was made. After the match was made, the test subject pressed a button to save their response and move on to the next trial. Each trial displayed only one Oz light, picked at random. Each Oz light was repeated for 5 trials, allowing us to measure any variability in the matches for a fixed Oz light.

The main limitation of this experiment was the small size of the test subject's classified region. The classified region is the area of the cone mosaic with known LMS identities. Our test subject had only a 0.5 $\times$ 0.5 degree area classified. Since Oz Vision depends on knowing the LMS identities, we can only present colors within a window that overlaps with this small 0.5 $\times$ 0.5 degree field-of-view. Eye drift makes this requirement even tighter, since this window needs to overlap with the classified region for the entire duration of the presentation. For this reason, we needed to restrict the Oz light to occupy a smaller 0.25 $\times$ 0.25 degree rectangle. The Oz light presentations were initialized such that they would start centered on the classified region. We also needed to restrict the duration to 0.5 seconds so that the classified region did not move out from under the Oz light. Additionally, the classified region is at 1.5 degrees eccentricity, meaning that the test subject needed to fixate at a target located 1.5 degrees away from where the actual colors were being presented. The uniform light was also displayed with the same size and duration as the Oz light to make matches more clear to the test subject. Future iterations should use test subjects with broader classified regions.

We also only had a single test subject available for this experiment. This test subject was previously involved in numerous psychophysics experiments using an AOSLO, and was therefore already well-trained as a test subject for these kinds of psychophysics procedures. This made data collection far less error-prone, but in future iterations we should include more test subjects as well as larger classified regions. We were unable to do this due to pandemic conditions.

Additionally, the current AOSLO system emits a constant "background light" even when Wizard modulates the laser to its lowest level. This background light normally appears as a dim red. To avoid biasing the color match results, we added complementary background light from a color projector in order to make the background appear a dim achromatic white. This background light caused both the Oz and uniform stimuli to be less saturated.

5.2 Color Match Results

The perceived Oz colors spanned a range of hues from a green to a yellow-orange. As expected, when attempting to show `LMS` $\begin{bmatrix}1 & 1 & 0\end{bmatrix}$, the test subject perceived the natural color of the 543 nm laser. When attempting to show `LMS` $\begin{bmatrix}1 & 0 & 0\end{bmatrix}$, the test subject reported seeing shades of yellow and yellow-orange.

However, this range of colors is far more compressed than we would have expected given Wizard's deliveries. We expected to see much stronger percepts of red when showing `LMS` $\begin{bmatrix}1 & 0 & 0\end{bmatrix}$. For the LMS presentations between those two endpoints, we similarly only observed colors between the two extremes of green and yellow-orange.

Using the color-match data and Wizard log files, we can quantitatively measure this level of compression, and offer one possible explanation. This data is shown in Figure 5.1. By running the EAVT on the log files for each presentation, we can calculate the arrived $L/(L+M)$ chromaticity to compare with the test subject's perceived chromaticity. The match chromaticities are much lower than the calculated arrived chromaticities, which is the compression effect. By tweaking the EAVT's parameters to fit the calculated arrived chromaticities to the match data, we can come up with an explanation for this compression. The simplest explanation is that the light arriving at the cone mosaic is more blurred than we expect, which would cause our fine spatial structure to break down. This in turn would pull the match colors closer to the natural color of the 543 nm stimulus beam. As shown in 5.1, a blur of $\sigma = 0.5$ arcminutes is sufficient to fit to the matches.

This additional blurring could itself be explained by a variety of factors:

- The EAVT's model for how microdoses translate into color percepts might be incomplete.
- The cone apertures might be broader than expected.
- The AOSLO might not be focusing the 543 nm light as well as we expect.

Although we were unable to show the full range of colors that we expected to be possible with Wizard, this early result is a major milestone in its development. In previous attempts, only the natural color of the 543 nm laser was visible, due to deficiencies in Wizard's cone-targeting accuracy. This color match data demonstrates that Wizard's engineering has finally achieved the level necessary to elicit color percepts using spatial structure alone.

5.3 A Shot in the Dark: Displaying an Impossible Color

Right after collecting the color-match measurements, we attempted to show the pure M stimulus through Wizard. This attempt was not to collect formal quantitative data, but just to record the test subject's qualitative experience with this stimulus. Although at

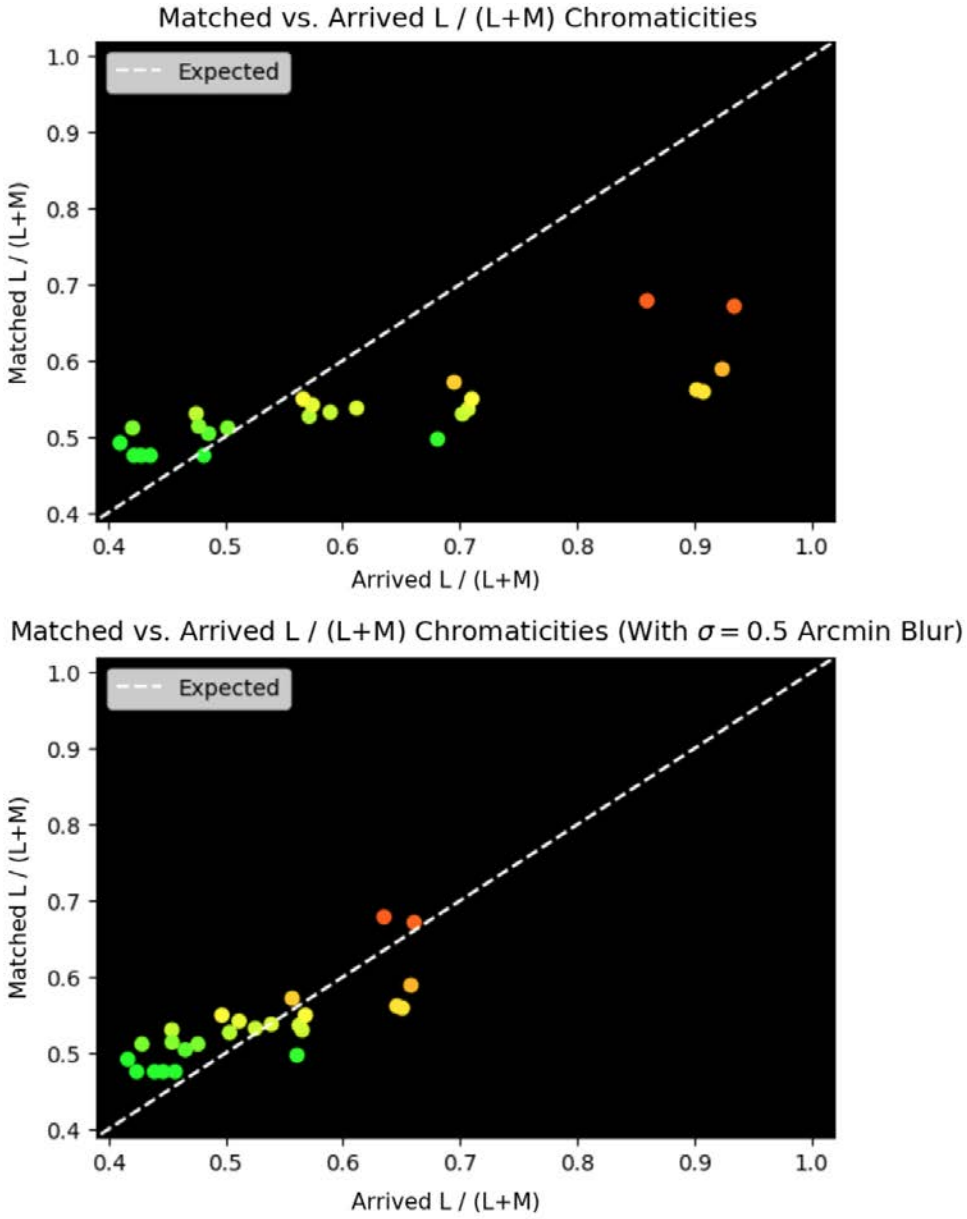

Figure 5.1: The matched chromaticities for different Oz presentations compared with the calculated arrived chromaticities. Top: Assuming no additional blurring. Bottom: Assuming that the light arriving at the retina had an additional Gaussian blur with $\sigma = 0.5$ arcminutes. The calculated chromaticities are generated from Wizard's log files describing how the 543 nm laser light was sent to the eye. The matched chromaticities are the responses of the test subject. The dashed white line shows where these points would lie if the calculated chromaticities and matches were in perfect agreement. The colors are a rough approximation of the test subject's perception.

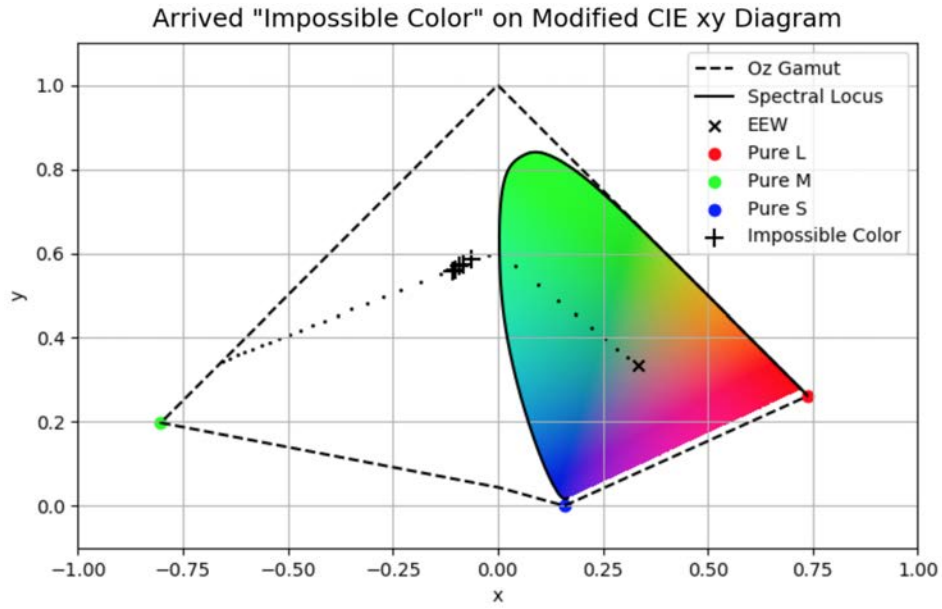

Figure 5.2: Estimated location of the pure M stimulus. The black crosses show where in modified CIE xy space our stimulus ended up landing. We intended to show a pure M stimulus, but delivery errors and background light cause the stimulus to get pulled toward the spectral gamut. The dotted line shows a straight line in LMS space extending out from equal energy white.

this point we were aware of the color compression, we expected that the pure M would be extreme enough to still be out of the spectral gamut. Showing this pure M stimulus was not the original goal of that experimental session, and so any conclusions drawn from this presentation are highly speculative.

The test subject described the stimulus as being a "highly saturated bluish green," while also having visible spatial structure. The test subject noted that the blue-green hue was distinct from the natural color of the 543 nm laser. The rectangular extent of the stimulus looked like a window through which they saw a texture of blue-green dots translating together. They also reported that these blue-green dots moved as though they were stuck to the retina, and suggested that they were seeing the structure of their cone mosaic. This test subject has experienced many AOSLO stimuli in previous experiments, and is familiar with that kind of motion.

Although we did not have a means to get an objective color match for this stimulus, we can use the log files to predict a chromaticity. Using the fitted EAVT model from figure 5.1, we arrive at the chromaticities shown in figure 5.2. Even with the $\sigma = 0.5$ blur, the predicted chromaticities are beyond the spectral gamut, and should correspond to impossible colors. The relative contribution of the background light for this presentation is not known. If the

background light is sufficiently bright, the perceived color could be "desaturated" back to the spectral gamut. However, even with a generous estimate[1] of that brightness, we are still in impossible color territory.

Although we are still not fully convinced that the perception really was the result of an impossible color, the appearance of both a blue tint and spatial structure not seen for the spectral gamut stimuli is intriguing on its own. Since there are many possible retinal images that yield the same cone activations, we might expect the visual system to make a Bayesian inference of the image given the cone activations. If this is true, perhaps the spatial structure is to be expected, since no uniform hue could have produced such a pattern of activity. However, it is unknown why the hue would also be noticeably more blue.

Regardless of the exact interpretation of this result, the fact that such inquiries can even be made demonstrates that this project has begun to scratch the surface of an unexplored region of human color perception, and will soon have the means to collect data to provide new answers.

[1]Figure 5.2 assumes that the background light is 1.5 times more luminous than the Oz light.

Chapter 6

Conclusion

In this thesis, we have successfully elicited a variety of color percepts using only a single wavelength of laser light. This Oz Vision method of color reproduction is unlike the traditional approach of mixing together lights of different hues. Our method sends a spatial distribution of light precise enough to isolate individual cone cells. With this precision, we elicit patterns of activity consistent with differently colored stimuli on a cone-by-cone basis. Since the neural signal leaving the retina entangles shape and color information, the visual system does not see the pattern of microdoses, but instead sees those different colors.

To achieve this level of control, we developed Wizard: a new software platform providing a per-photoreceptor graphics display. Wizard is currently built as a software control layer on top of an adaptive optics scanning laser ophthalmoscope (AOSLO). The AOSLO is the optics hardware capable of simultaneously imaging the retina using an 840 nm laser and stimulating the retina with a 543 nm laser. Wizard uses this live video of the retina to track the eye's motion to high precision with low latency, and modulates the 543 nm laser to send microdoses to individual cone cells. By varying the intensity of these microdoses, we can directly control the activity levels of cones within the patch of retina visible to the AOSLO. Thus, Wizard provides a display with photoreceptors as the fundamental display element, rather than pixels.

To be a useful display technology, Wizard allows external software to control what graphics are presented to the cone cells. This was done by exposing a set of commands to Matlab. For example, they can change what LMS tristimulus value is shown or orchestrate Wizard's behavior in conjunction with other external programs. These Matlab commands provide users with a platform to display per-photoreceptor graphics without needing to dig into Wizard's internal code. We put these commands into practice by implementing our color match experiment on top of them.

Wizard also keeps a detailed record of what was sent out to the eye in order to quantify any errors in delivery. This record provided the necessary context for interpreting the color match results. Without these log files, we would have only known what tristimulus value was intended, and not what actually arrived. Using the EAVT on the microdose log files, we computed what tristimulus value actually arrived at the test subject's retina. By fitting

the EAVT's output to the test subject's color matches, we obtained a model for predicting chromaticities given Wizard's record of what was sent to the eye.

This model predicted that we successfully sent tristimulus values with chromaticities outside of the spectral gamut, even given assumptions that there is more blur and background light than we expect. If this prediction is true, then the test subject's report of a non-spatially uniform "highly saturated bluish green" provokes many new scientific inquiries. Is this spatial structure a general feature of out-of-gamut deliveries? Where does the blue tint come from? Could a test subject eventually condition themselves to see this as an impossible color? Future work to probe into these phenomena is already within reach.

By directly modifying the activity levels of the cells responsible for color vision, we have manipulated color perception at its most fundamental level. With this level of control, we can begin to probe deeper into the earliest stages of human vision. In particular, it has driven serious discussion on the possibility of "impossible" colors arising from tristimulus values outside of the spectral gamut. It is through this possibility that the Oz Vision display may someday *expand* the range of human perception, not just *reproduce* it.

Bibliography

[1] David W. Arathorn et al. "How the unstable eye sees a stable and moving world". eng. In: *Journal of vision* 13.10 (Aug. 2013). 13.10.22[PII], pp. 22–22. ISSN: 1534-7362. DOI: 10.1167/13.10.22. URL: https://doi.org/10.1167/13.10.22.

[2] Arthur G. Bennett and Ronald B. Rabbetts. *Clinical visual optics*. English. London; Boston: Butterworths, 1984.

[3] David Brainard and Andrew Stockman. "Colorimetry". In: *The Optical Society of America Handbook of Optics, 3rd edition, Volume III: Vision and Vision Optics*. Ed. by Michael Bass et al. McGraw-Hill, 2010, pp. 11.1–11.104.

[4] *CIE (2012) 2-deg XYZ "physiologically-relevant" colour matching functions*. URL: http://www.cvrl.org/database/text/cienewxyz/cie2012xyz2.htm.

[5] Christine Curcio et al. "Human photoreceptor topography". In: *The Journal of comparative neurology* 292 (Feb. 1990), pp. 497–523. DOI: 10.1002/cne.902920402.

[6] Martin A. Fischler and Robert C. Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". In: *Commun. ACM* 24.6 (June 1981), pp. 381–395. ISSN: 0001-0782. DOI: 10.1145/358669.358692. URL: https://doi.org/10.1145/358669.358692.

[7] Erich Gamma et al. *Design Patterns: Elements of Reusable Object-Oriented Software*. USA: Addison-Wesley Longman Publishing Co., Inc., 1995. ISBN: 0201633612.

[8] Wolf M. Harmening et al. "Mapping the Perceptual Grain of the Human Retina". In: *Journal of Neuroscience* 34.16 (2014), pp. 5667–5677. ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.5191-13.2014. eprint: https://www.jneurosci.org/content/34/16/5667.full.pdf. URL: https://www.jneurosci.org/content/34/16/5667.

[9] Jan J. Koenderink. *Color for the Sciences*. The MIT Press, 2010. ISBN: 0262014289.

[10] R. John Leigh and David S. Zee. *Neurology of Eye Movements*. Oxford University Press, 2015.

[11] J.P. Lewis. "Fast Normalized Cross-Correlation". In: *Ind. Light Magic* 10 (Oct. 2001).

[12] Kaccie Y. Li and Austin Roorda. "Automated identification of cone photoreceptors in adaptive optics retinal images". In: *J. Opt. Soc. Am. A* 24.5 (May 2007), pp. 1358–1363. DOI: 10.1364/JOSAA.24.001358. URL: http://josaa.osa.org/abstract.cfm?URI=josaa-24-5-1358.

[13] Donald I.A. Macleod, David R. Williams, and Walter Makous. "A visual nonlinearity fed by single cones". In: *Vision Research* 32.2 (1992), pp. 347–363. ISSN: 0042-6989. DOI: https://doi.org/10.1016/0042-6989(92)90144-8. URL: https://www.sciencedirect.com/science/article/pii/0042698992901448.

[14] Susana Martinez-Conde, Stephen L. Macknik, and David H. Hubel. "The role of fixational eye movements in visual perception". In: *Nature Reviews Neuroscience* 5.3 (Mar. 2004), pp. 229–240. DOI: 10.1038/nrn1348. URL: https://doi.org/10.1038/nrn1348.

[15] John Nickolls et al. "Scalable Parallel Programming with CUDA: Is CUDA the Parallel Programming Model That Application Developers Have Been Waiting For?" In: *Queue* 6.2 (Mar. 2008), pp. 40–53. ISSN: 1542-7730. DOI: 10.1145/1365490.1365500. URL: https://doi.org/10.1145/1365490.1365500.

[16] Austin Roorda et al. "Adaptive optics scanning laser ophthalmoscopy". In: *Opt. Express* 10.9 (May 2002), pp. 405–412. DOI: 10.1364/OE.10.000405. URL: http://www.opticsexpress.org/abstract.cfm?URI=oe-10-9-405.

[17] Michele Rucci, Ehud Ahissar, and David Burr. "Temporal Coding of Visual Space". In: *Trends in cognitive sciences* 22 (Oct. 2018), pp. 883–895. DOI: 10.1016/j.tics.2018.07.009.

[18] W. A. Rushton. "Pigments and signals in colour vision". eng. In: *The Journal of physiology* 220.3 (Feb. 1972). PMC1331666[pmcid], 1P–31P. ISSN: 0022-3751. DOI: 10.1113/jphysiol.1972.sp009719. URL: https://doi.org/10.1113/jphysiol.1972.sp009719.

[19] Christy K. Sheehy et al. "High-speed, image-based eye tracking with a scanning laser ophthalmoscope". eng. In: *Biomedical optics express* 3.10 (Oct. 2012). 173117[PII], pp. 2611–2622. ISSN: 2156-7085. DOI: 10.1364/BOE.3.002611. URL: https://doi.org/10.1364/BOE.3.002611.

[20] Lawrence C. Sincich et al. "Resolving single cone inputs to visual receptive fields". In: *Nature Neuroscience* 12.8 (Aug. 2009), pp. 967–969. ISSN: 1546-1726. DOI: 10.1038/nn.2352. URL: https://doi.org/10.1038/nn.2352.

[21] Scott B. Stevenson and Austin Roorda. "Correcting for miniature eye movements in high resolution scanning laser ophthalmoscopy". In: *Ophthalmic Technologies XV*. Ed. by Fabrice Manns et al. Vol. 5688. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series. Apr. 2005, pp. 145–151. DOI: 10.1117/12.591190.

[22] Andrew Stockman and Lindsay T. Sharpe. "The spectral sensitivities of the middle- and long-wavelength-sensitive cones derived from measurements in observers of known genotype". In: *Vision Research* 40.13 (2000), pp. 1711–1737. ISSN: 0042-6989. DOI: https://doi.org/10.1016/S0042-6989(00)00021-3. URL: https://www.sciencedirect.com/science/article/pii/S0042698900000213.

[23] Yiyi Wang et al. "Human foveal cone photoreceptor topography and its dependence on eye length". eng. In: *eLife* 8 (July 2019). 47148[PII], e47148. ISSN: 2050-084X. DOI: 10.7554/eLife.47148. URL: https://doi.org/10.7554/eLife.47148.

[24] D. Williams et al. "How Far Can We Extend the Limits of Human Vision ?" In: 2003.

Appendix A

Calculating Chromaticities from LMS Tristimulus Values

A.1 Maxwell's Color Triangle

Given a tristimulus value in LMS color space, we can calculate the lm chromaticity coordinates on Maxwell's color triangle as follows:

$$l = L/(L + M + S)$$
$$m = M/(L + M + S)$$
$$s = S/(L + M + S)$$

Since $l + m + s = 1$, we only need the first two dimensions, lm, to define a chromaticity.

Maxwell's color triangle shows the barycentric coordinates of an underlying 3D space. The triangle was not originally applied to LMS space, but instead to the mixtures of three real-world colors.

A.2 Modified CIE xy

Modified CIE xy is a variation of the "physiologically-relevant" CIE 2006 provided at [4]. The modification is to take the absolute value of the X, Y, and Z elements when performing the perspective divide.

Given a tristimulus value in LMS color space, we can calculate the modified CIE xy chromaticity coordinates as follows:

$$x = X/(|X| + |Y| + |Z|)$$
$$y = Y/(|X| + |Y| + |Z|)$$
$$z = Z/(|X| + |Y| + |Z|)$$

Where:

$$X = 1.94735469L - 1.41445123M + 0.36476327S$$

$$Y = 0.68990272L + 0.34832189S$$

$$Z = 1.93485343S$$