VISUAL IMAGE SIMULATION (VIS) TOOL
Design Description Document
DigitEyez / Brandon Zimmerman

Project Coordinator: Ali Hashim
Customer Liaison: Diego Martinez
Document Handler: Perry Wang
Scribe: Weidi Liu
Faculty Advisor: Dr. Jennifer Hunter

Document Number 08

Revisions Level	Date
H	5-04-2018

This is a computer-generated document. The electronic master is the official revision. Paper copies are for reference only. Paper copies may be authenticated for specifically stated purposes in the authentication block.	Authentication Block

Confidential Information Subject to Non-Disclosure Agreement

Rev	Description	Date	Authorization
A	Initial Document	1-22-2018	PBW
B	Document 2	2-07-2018	AH
C	Document 3	2-20-2018	AH
D	Document 4	2-25-2018	PBW
E	Midterm Document	3-01-2018	AH
F	Document 6	4-08-2018	AH
G	Document 7	4-24-2018	AH
H	Final Document	5-04-2018	AH

Table of Contents

Revision History **2**

Vision Statement **4**

Project Scope **4**

Assumptions **4**

Results **4**

- Features of the MATLAB VIS Tool 4
- Code V Model 6
- Gradient Letter Chart 7
- Zernike Polynomials 8
- ABCD Matrix 8
- Pupil Scaling 9
- Stiles-Crawford Effect 9
- Polychromatic PSF 9
- Image Quality Metrics 10
- Patient Data 11

Predictive Prescription Algorithm Proposal **13**

Future Improvements **14**

Design Day **15**

Schedule **15**

References **16**

Appendices **17**

Special Thanks **22**

The Visual Image Simulation tool is a software tool that will be used in conjunction with DigitEyez's proprietary algorithm, spearheaded by Brandon Zimmerman, and was developed with the help of our faculty advisor, Dr. Jennifer Hunter.

Vision Statement:

The vision for this project was to develop a digital visualization tool for DigitEyez to simulate how a patient's eyes will perform in a routine eye exam. The goal is to use this tool to assist in providing accurate prescriptions for eye care while reducing the time, cost, and human error currently associated with the common exam routine. The Visual Image Simulation (VIS) tool will specifically aid ophthalmologists and optometrists to give more accurate prescriptions, whereas other self-prescribing apps on the market are only used by the consumers. The VIS tool will be used in conjunction with DigitEyez's proprietary algorithm, which will directly determine the ideal prescription of the patient using only autorefractor measurements within ¼ of a diopter. The patient's subjective feedback from using the VIS tool will be used to confirm that the predicted prescriptions are correct, or if they need to be modified.

Project Scope:

A final visual image simulation tool was coded in MATLAB, and the lens model was produced in Code V to compare the PSF of our VIS tool via Code V's real ray tracing algorithm. We were not responsible for determining the prescription of the patient, obtaining patient data, or the numerical ray trace. We were not responsible for determining the Zernike coefficients of the patient's eye, as they were given to us based on assumptions made in DigitEyez's proprietary algorithm about the properties of the eye. There was no cost for this project since it was developed using free software.

Assumptions:

It is assumed that the images will be simulated for a patient's vision at room temperature and normal humidity. 555 nm is the wavelength that is used as the focus wavelength. To make the VIS polychromatic, we used the spectral luminous efficiency to weigh the different wavelengths to obtain a polychromatic PSF. The distance that the patient's vision is simulated from is from 20 feet away from the image. The VIS tool is used to provide retinal images for each of the patient's eyes.

Results:

Features of the MATLAB VIS tool:

1. Produces a retinal image using Zernike coefficients from a patient.
2. Takes into account the apodization of the eye to mimic the Stiles-Crawford effect at the retina.
3. The VIS tool can produce a polychromatic PSF. The PSF is produced for each wavelength in its respective color channel (red, green, or blue) and is weighted by the spectral luminous efficiency of the eye. The red, green, and blue weighted PSFs are then combined to produce an RGB image.

4. Accounts for the toric nature of the cornea by allowing the use of an elliptical-pupil function instead of a circular-pupil function. Elliptical Zernike polynomials can also be used instead of circular Zernike polynomials.
5. Provides image quality metrics including the Visual Strehl Ratio which applies a neural weighting function, and another image quality metric based off of the correlation of two images, found through a convolution.
6. Computes an ABCD matrix based off of the Diaz paper to find the effective focal length of the crystalline gradient index of the eye.

During the fall semester we used Zernike coefficients from Watson's paper, "Computing Human Optical Point Spread Functions" to compute the PSF and convolved it with a target image to produce a retinal image using a circular pupil.

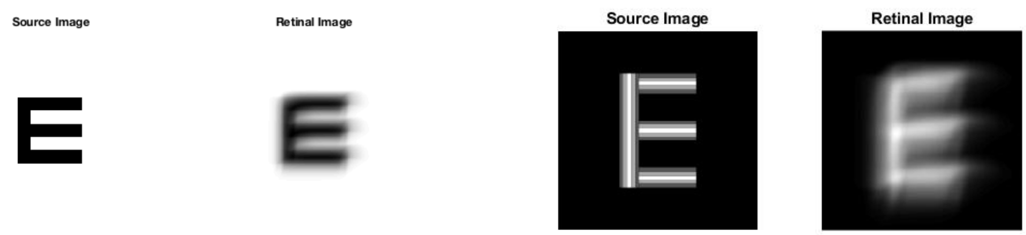


Figure 1: Images taken from our current Matlab algorithm depicting a source image on the left and a retinal image on the right using hard-coded Zernike polynomials and a circular pupil, for each a regular Snellen E and a gradient-letter E.

This semester, we have integrated Stiles Crawford effect, pupil scaling, refractive Zernike polynomials, and image quality metrics into our software. We have also completed a polychromatic PSF function based on a weighted RGB spectrum. Additionally, we have created a Code V model, as a sanity check, using a perfect lens to model the PSF given inputted Zernike polynomials. A ray transfer matrix analysis was used to reproduce the GRIN lens model. A simple iOS application was developed that can be easily integrated with our VIS tool to display the images on an iPhone or iPad.

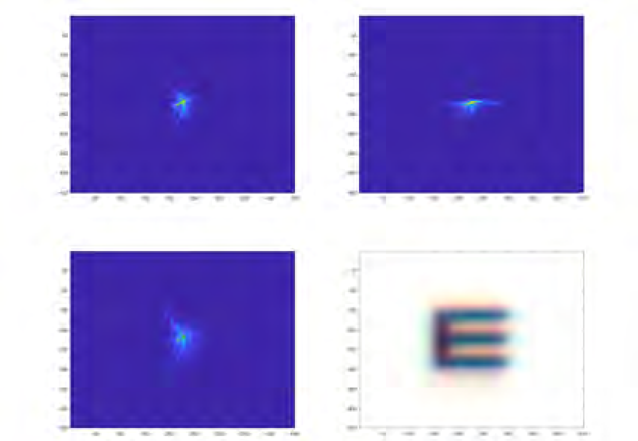


Figure 2: PSF for RGB colors, and an image taken from our current Matlab algorithm showing a retinal image using hard coded Zernike polynomials and a circular pupil with chromatic aberration. Top left to bottom right: red PSF, green PSF, blue PSF, and retinal chromatic image.

Figure 3: Screenshots of the iOS application we created for design day. The application uses a pickerview to scroll through different patients and three different image types to show the uncorrected and corrected retinal images in grayscale and in color.

Code V Model:

We produced an optical design of the Liou & Brennan eye model in Code V using spherical surfaces. The purpose of the model is to check the PSF and image results from our VIS tool. The gradient index and other factors of this eye model can be adjusted based on the autorefractor measurements of the patient's attributes.

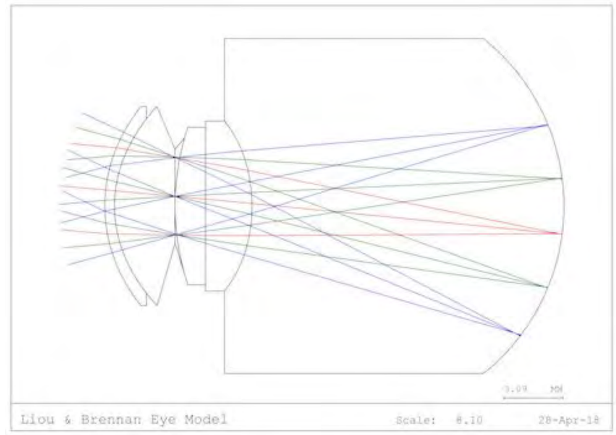


Figure 4: Liou and Brennan eye model that we produced in Code V.

A perfect lens was also modelled with Code V to confirm that the PSFs produced from the VIS tool are consistent with that produced in Code V. The monochromatic PSFs are shown below and demonstrates that the VIS tool computes the PSF correctly:

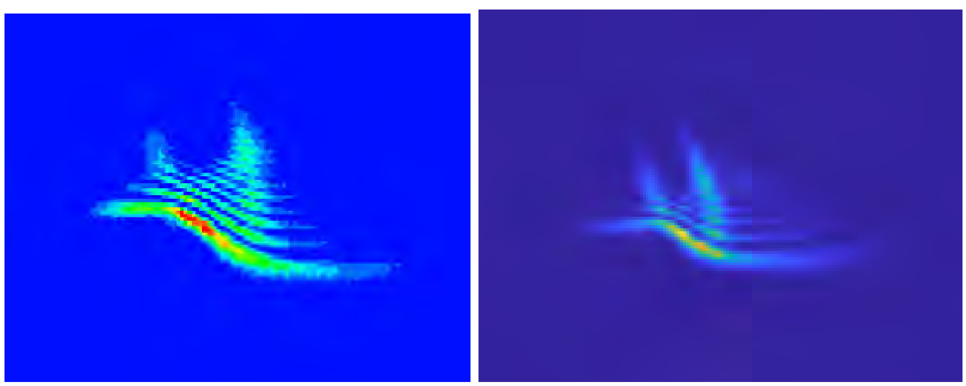


Figure 5: PSF found using perfect lens with hard-coded Zernike coefficients model in Code V (left) and PSF found using Matlab VIS tool (right). Both PSFs are inverted due to the negative magnification.

Gradient Letter Chart

We have produced a gradient letter chart using Adobe Illustrator and a .otf file for DigitEyez's work with Ben Vision Research. Users can install the font and use it in a software program, which enables simple image resizing so that images with a size of 20/20 and 20/40 can easily be produced. The purpose of the gradient letter chart as opposed to a regular Snellen letter chart is that the number of lines in the image that the patient can see contributes to the exam. The number of lines a patient can observe provides more information than a regular eye exam.

Figure 6: Gradient letter chart created using Adobe Illustrator that will be used for the visual image simulation tool. From top to bottom: 20/200, 20/100, 20/70, 20/50, 20/40, 20/30, & 20/20. Not to scale.

Zernike Polynomials

The VIS tool allows DigitEyez to represent the aberrations of patient's eyes with standard circular and elliptical Zernike polynomials. The VIS tool also enables DigitEyez to use refractive Zernike polynomials instead of fringe Zernike polynomials. DigitEyez currently represents the aberrations of patient's eyes with refractive Zernike polynomials and this model will be easier to integrate with their codebase.

ABCD Matrix

An ABCD matrix representation for the Gullstrand eye model has been implemented to improve computational efficiency and integration of DigitEyez's predictive prescription algorithm. The current algorithm and constants used for calculating the Gullstrand exact eye model ABCD matrix is based on Chapter 2 of *Modeling the Optical and Visual Performance of the Human Eye*. The final results produced is close to the results presented in the Diaz paper, and any differences are due to different constants assumed for different patients. For the Code V model, a gradient-index lens was inputted in the Liou & Brennan eye model.

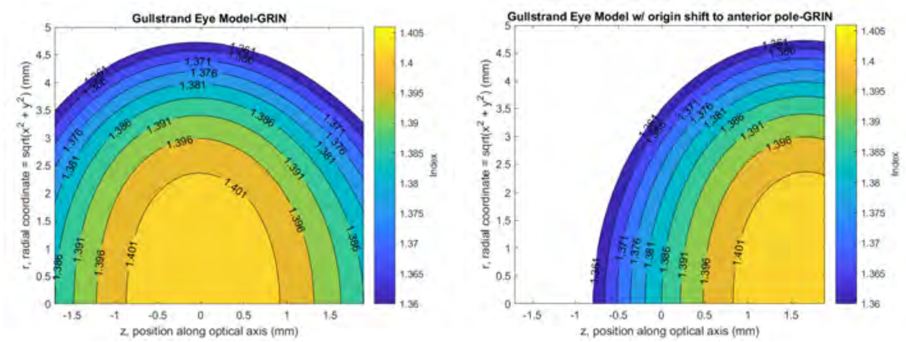


Figure 7: Plots of the GRIN profile with index increments of 0.005 as a function of z, the position along the optical axis, and r, the radial distance from the optical axis. On the left, the general formula of the Gullstrand GRIN Eye Model was used, and on the right is the same formula, but with a shift in z.

The plots of the GRIN profile were produced to help understand the formulas being used in the ABCD matrix calculations. They also validated the work done in MATLAB, since these plots are the same a plot of the GRIN profile showed in Chapter 2 of *Modeling the Optical and Visual Performance of the Human Eye.*

Pupil Scaling

The retinal image that is outputted by the VIS tool is currently scalable to microns from pixels. The unit scaling can be compared for different source image sizes from a letter chart. Additionally, unit scaling is implemented with the polychromatic retinal image.

Stiles-Crawford Effect

The Stiles-Crawford Effect is the nonuniformity of light from the pupil reaching the retina. It can be represented by the relationship below.

$$\eta(d) = \eta(d_m)\ 10^{-p(\lambda)(d-d_m)^2}$$

Where η is the relative luminance efficiency, d is the distance away from the center of the pupil, d_m is the average largest symmetric distance from the center of the pupil, and $p(\lambda)$ is the wavelength dependent magnitude of the pupil.

We mapped this Gaussian function onto the complex pupil function. The Stiles-Crawford effect includes the wavelength-dependent parameter labelled as $p(\lambda)$, which is used for the polychromatic retinal image.

Polychromatic PSF

The polychromatic PSF takes into account the lateral chromatic aberrations of the eye. We weighted the PSF's for each of red, green, and blue channel based on spectral sensitivity data taken of the human eye. The details of the implementation are specified in the appendix.

Image Quality Metrics

Finally, there are quantifiable metrics to assess the retinal images produced from the VIS tool. The Visual Strehl Ratio is a way to capture the effectiveness of a PSF for stimulating the neural portion of the visual system. It is an inner product of the PSF with a neural weighting function normalized to the diffraction-limited PSF. The neural weighting function is the inverse Fourier transform of the neural contrast sensitivity function. The Visual Strehl function is a Gaussian function with a phase equal to the sum of the Zernike polynomials.

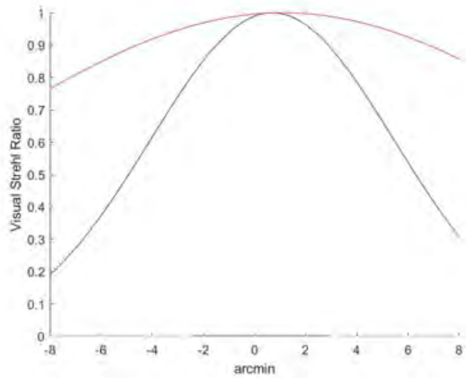


Figure 8: Visual Strehl Ratio of Patient A without prescription (black) and with prescription (red). The Visual Strehl Ratio indicates how sensitive a patient's vision is to different spatial frequencies. The Visual Strehl Ratio computed with the prescription for patient A indicates that their vision is less sensitive to spatial frequencies as they are without the prescription.

The image convolution metric is another image quality metric that compares the retinal image with the source image. It computes the correlation between the two images by finding the maximum value of their convolution. The image convolution metric is a good technique for determining how good corrective prescriptions are because the correlation can be computed for different amounts of defocus and astigmatism. This technique was formalized by Len Zheleznyak.

Figure 9: Retinal images computed with different amounts of defocus, with the image convolution correlation computed for each one

The correlations show that as the defocus changes, the correlation decreases. This will be a valuable tool to optimize the defocus of a predicted prescription to maximize the correlation between the source and retinal images. The implementation of the features of the VIS tool are explained in more detail in the appendix.

Current Patient Data provided by Digiteyez

Patient	Gender	Age	Focal Length (mm)
A	M	30	13.302
B	F	28	14.7393
C	M	23	16.3513
D	F	29	16.3513

Table 1: Focal lengths for sample patients provided by DigitEyez.

Brandon Zimmerman provided us with Zernike coefficients of four patients before and after they received prescriptions. The table above shows the details of each patient, including the focal length of their eye, as evaluated by Digiteyes. We took the accompanying Zernike coefficients for each patient, and ran them through the Grayscale.m MATLAB function. The Zernike coefficients from DigitEyez can be found in PatientData.m. For our propagations, we scaled the weights of the Zernike coefficients before prescription to sum to 1, since the Zernike coefficients DigitEyez provided were not scaled. These scaled Zernike coefficients were saved in PatientData_scaled.m. The PSF functions and the retinal images for patients were saved, for patients with and without prescription. Figure 8 and Figure 9 show these images.

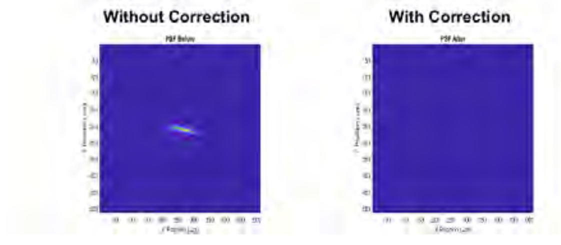
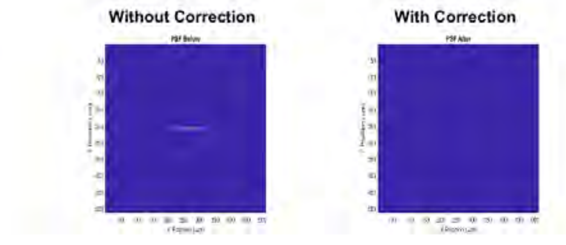
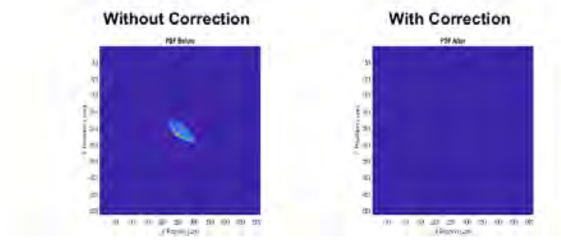
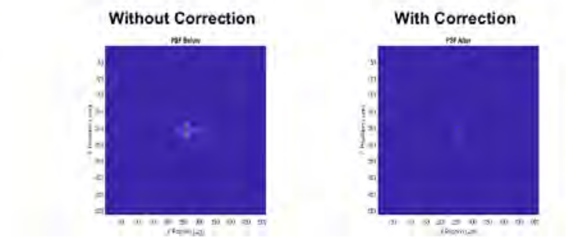
Patient A	**Patient B**
	
Patient C	**Patient D**
	

Figure 10: PSF images computed for the four patients' Zernikes that DigitEyez provided before and after prescription.

The PSFs show that the corrective prescriptions help minimize the aberrations for each patient. However, for patient D, who has larger higher-order aberrations than the other patients, the corrective prescription minimizes the aberrations less than it does for the other patients.

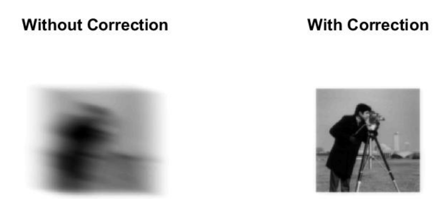
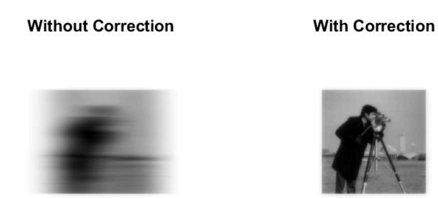
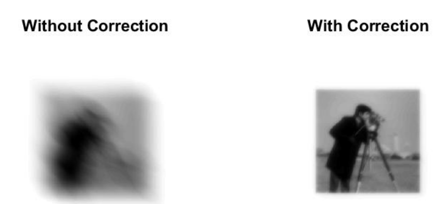
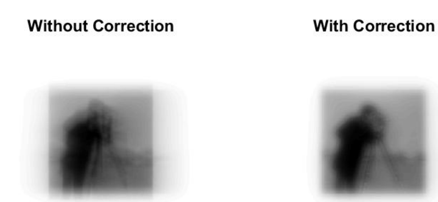
Patient A	**Patient B**
	
Patient C	**Patient D**
	

Figure 11: Retinal images computed for the four patients' Zernikes that DigitEyez provided before and after prescription

The patient's Zernike coefficients were also inputted to Polychromatic.m, and using a Snellen letter E and a gradient letter E in addition to the image of the cameraman. See Appendix D to see these images as well.

Predictive Prescription Algorithm Proposal

DigitEyez has struggled to implement a predictive prescription algorithm that performs well on novel data. In an effort to address this, we have detailed a machine learning algorithm that would predict a patient's prescription.

A decision tree is a form of supervised learning that requires a dataset with known prescriptions of patients with their autorefractor measurements, previous prescription, age, gender, and other demographic information. The dataset will be split into a test set and a training set. The training set will be used to build a decision tree. A decision tree reaches a decision by performing a sequence of tests. At each internal node in the tree, a value corresponding to one of the input attributes is stored. Each leaf node in the tree specifies a value to be returned by the function.

Example	Input Attributes										Goal
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
$\mathbf{x}_1$	Yes	No	No	Yes	Some	$$$	No	Yes	French	0–10	y_1 = Yes
$\mathbf{x}_2$	Yes	No	No	Yes	Full	$	No	No	Thai	30–60	y_2 = No
$\mathbf{x}_3$	No	Yes	No	No	Some	$	No	No	Burger	0–10	y_3 = Yes
$\mathbf{x}_4$	Yes	No	Yes	Yes	Full	$	Yes	No	Thai	10–30	y_4 = Yes
$\mathbf{x}_5$	Yes	No	Yes	No	Full	$$$	No	Yes	French	>60	y_5 = No
$\mathbf{x}_6$	No	Yes	No	Yes	Some	$$	Yes	Yes	Italian	0–10	y_6 = Yes
$\mathbf{x}_7$	No	Yes	No	No	None	$	Yes	No	Burger	0–10	y_7 = No
$\mathbf{x}_8$	No	No	No	Yes	Some	$$	Yes	Yes	Thai	0–10	y_8 = Yes
$\mathbf{x}_9$	No	Yes	Yes	No	Full	$	Yes	No	Burger	>60	y_9 = No
$\mathbf{x}_{10}$	Yes	Yes	Yes	Yes	Full	$$$	No	Yes	Italian	10–30	y_{10} = No
$\mathbf{x}_{11}$	No	No	No	No	None	$	No	No	Thai	0–10	y_{11} = No
$\mathbf{x}_{12}$	Yes	Yes	Yes	Yes	Full	$	No	No	Burger	30–60	y_{12} = Yes

Figure 18.3 Examples for the restaurant domain.

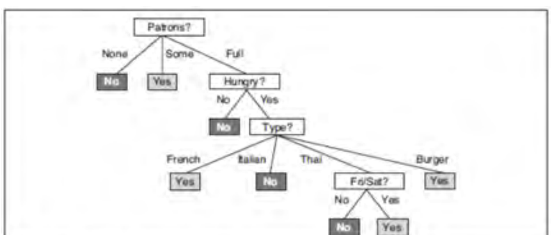


Figure 18.6 The decision tree induced from the 12-example training set.

Figure 12: Table (above) depicting a sample training dataset for a problem of whether someone should wait or not at a restaurant to be seated. The decision tree (below) shows the decision tree that is generated from this dataset that explains how someone should make his or her decision. This is from Stuart Russell's Artificial Intelligence textbook.

The table above depicts a sample dataset, with a target class of figuring out whether you should wait to be seated at a restaurant. The first internal node that is used to split the data is based on which input attribute leads to the most consistent goal label. In other words, the more important attributes that lead to a classification of the goal are closer to the top of the tree. For example the patrons attribute is selected as the first node because whenever the patrons category = None, the goal class always yields no, and when the patrons category = Some, the goal class always equals yes. The attributes keep splitting by the input attribute conditioned on the parent attribute until you can always reach a decision.

Implementation for DigitEyez:

Input Attributes: Autorefractor measurements, previous prescription (Defocus, Astigmatism), age, gender, demographic, etc. (More the better)

Goal: Prescription (Defocus, Astigmatism)

Important note: For a decision tree, the data needs to be discretized. Age ranges need to be split such as (10…20, 21…30, etc.). Prescription labels should be vectorized from -15:15 diopters in 0.25 diopter increments. The decision tree algorithm should be repeated separately for defocus and astigmatism. Based off of initial analysis, the most important attribute to determining a patient's prescription is his or her previous prescription. Prescriptions do not change by a large factor, especially if the time between eye exams is short. If the patient has not been prescribed before and is younger, his or her vision would be harder to prescribe using this algorithm because the variance of the dataset is larger.

Advantages

One can visually understand how the decision tree made its decision, which is not possible via a neural network. You can prune based on prior knowledge. In our case, a patient's prescription should be close to their last prescription. If there is no prescription, it can be its own label for the previous prescription class. Avoids overfitting the dataset.

Disadvantages:

Discretization: the data needs to be separated into categories for it to work. Also, if there are patients that have the same input attributes, but different prescriptions, it reduces the confidence level that the algorithm has to make its decision.

Future Improvements

The VIS tool can be improved by pairing it with DigitEyez's working predictive prescription algorithm. Right now, the algorithm uses Zernike coefficients provided from DigitEyez's algorithm, but doesn't seamlessly integrate with it. Additionally, due to the lack of patient data that DigitEyez has accumulated, their algorithm overfits the training data, and performs poorly on new patient data. To avoid this issue, DigitEyez should collect as much data as possible and implement a machine learning algorithm to predict a patient's prescription. Machine learning algorithms such as the one proposed can perform better with more data and can explain how it reached its decision.

For the polychromatic retinal image, the scaling from pupil samples to PSF samples only uses one wavelength. However, this unit scaling could be improved if the scaling interpolates over all the wavelengths that are used in the weighted spectrum.

Furthermore, the patient data received from DigitEyez was older data that has already been analyzed by our customer. In order to further test and verify the accuracy of our visual image simulation, it would be best to work with patients in person to receive their feedback when using our software. With their feedback, we could continue to adjust our product to increase its accuracy and improve its features.

Another idea that might make the predictive prescription algorithm more accurate is to construct a tool that works like an eye wavefront sensor to measure the Zernike coefficients of a patient without the large cost that comes with commercial wavefront sensors.

Design Day Demonstration

For design day, we demonstrated an iPad application that uses the VIS tool to produce retinal images with and without corrective prescriptions using a few sample patients. The simulated images are a normal Snellen letter, a gradient Snellen letter, and a real-world image (e.g. a cameraman picture) in grayscale and in color. A poster was created to discuss our project in more detail. Screenshots of the iOS application are shown in Figure 2.

Schedule:

January

19: Met with Dr. Hunter to discuss initial progress and immediate priorities.

31: Met with Brandon Zimmerman to discuss progress and to obtain access to pertinent components from DigitEyez's proprietary algorithm.

Deliverables: Incorporated Stiles-Crawford effect in VIS tool; finished Huygens-Fresnel propagation; began polychromatic model based on Watson's paper.

February

2: Met with Dr. Hunter to discuss reference papers and polychromatic PSF.

16: Met with Brandon Zimmerman to sign NDA.

16: Met with Dr. Hunter to sign NDA and to discuss current progress.

Deliverables: Polychromatic PSF set to RGB color channels was produced. We incorporated lateral chromatic aberrations.

March

2: Midterm design review.

2: Met with Dr. Hunter.

5: Met with Brandon Zimmerman.

23: Met with Dr. Hunter.

26: Met with Brandon Zimmerman.

Deliverables: Add corrective prescription lens to the algorithm; finished all requirements of the VIS tool; added any extra features.

April

13: Met with Dr. Hunter.

13: Met with Brandon Zimmerman.

15: Met with Len Zhelznyak

17: Met with Geunyoung Yoon

Deliverables: Added image convolution metric and Visual Strehl Ratio metrics; pupil scaling and spectral luminous efficiency weighting; Integrated the VIS tool into a mobile application; Documented and refactored code

May

4: Design Day

Deliverables: Presented an iPad app that has integrated the VIS tool.

References:

Abraham, Chandler. “A Beginner's Guide to (CIE) Colorimetry – Color and Imaging – Medium.” Medium, Augmenting Humanity, 10 Sept. 2016, medium.com/hipster-color-science/a-beginners-guide-to-colorimetry-401f1830b65a.

Campbell, F. W., & Green, D. G. (1965). Optical and retinal factors affecting visual resolution. Journal of Physiology, 181, 576-593.

Gobbi, Pier Giorgio. “Modeling the Optical and Visual Performance of the Human Eye”, SPIE Press. 2013

Huang, Yanqiao. “Human Lens Modeling and Biometric Measurement Technique.” Department of Biomedical Engineering, University of Rochester. 2008. http://hdl.handle.net/1802/6275

Iskander, D. Robert, et al. “Objective Refraction from Monochromatic Wavefront Aberrations via Zernike Power Polynomials.” *Ophthalmic and Physiological Optics*, vol. 27, no. 3, 2007, pp. 245–255., doi:10.1111/j.1475-1313.2007.00473.x.

J. A. Díaz, “ABCD matrix of the human lens gradient-index profile: applicability of the calculation methods,” Appl. Opt. 47, 195–205 (2008).APOPAI0003-6935 http://dx.doi.org/10.1364/AO.47.000195

Krešimir Matković. Tone Mapping Techniques and Color Image Difference in Global Illumination. 2003

L. N. Thibos et al, "Accuracy and precision of objective refraction from wavefront aberrations," Journal of Vision, vol. 4, (4), pp. 329, 2004.

Lindbloom, Bruce Justin. “RGB/XYZ Matrices.” Bruce Lindbloom's Website, 7 Apr. 2017, www.brucelindbloom.com/index.html?Eqn_RGB_XYZ_Matrix.html.

Schwiegerling, Jim. "Scaling pseudo-Zernike expansion coefficients to different pupil sizes," Opt. Lett. 36, 3076-3078 (2011)

Stuart Russell and Peter Norvig. 2009. Artificial Intelligence: A Modern Approach (3rd ed.). Prentice Hall Press, Upper Saddle River, NJ, USA.

Watson, Andrew B. “Computing Human Optical Point Spread Functions.” *Journal of Vision*, The Association for Research in Vision and Ophthalmology, 1 Feb. 2015, jov.arvojournals.org/article.aspx?articleid=2213266.

Williams, D. R. (2003). Subjective image quality metrics from the wave aberration. Paper presented at the 4th International Congress of Wavefront Sensing and Aberration-Free Refractive Correction, San Francisco, CA.

Williams, David R. "Sharpness Metric For Vision Quality," U.S. Patent 7,077,522, issued July 18, 2006

Zheleznyak, L. Overcoming Presbyopia by Manipulating the Eyes' Optics. 2014.

Appendices:

Appendix A: Basic Algorithm

$$w(x', y') = \sum_{n,m} c_n^m z_n^m(x', y') \quad (1)$$

where w(x', y') is the wavefront error function, which can be determined by summing up the Zernike polynomials with a given order n, given frequency m, and weighting coefficient c.

$$g(x', y') = p(x', y') exp(i(2\pi/\lambda)\, w(x', y')) \quad (2)$$

where g(x', y') is the generalized complex pupil function, p(x', y') is the pupil, and λ is the wavelength.

$$h(x, y) = \|F[g(x', y')]\|^2 \quad (3)$$

where h(x, y) is the PSF for incoherent light. It is given by the squared modulus of the Fourier transform of the generalized pupil function.

$$r(x, y) = h(x, y) * s(x, y) \quad (4)$$

where r(x, y) is the retinal image function. It is the the convolution of the source image s(x,y) and the PSF.

Appendix B: Chromatic Aberrations

The lateral chromatic aberrations that are used in our VIS tool are computed via the equations below which are formalized from the Watson paper.

$$D_{589}(\lambda) = 1.68524 - 0.63346/(\lambda - 0.2141) \quad (5)$$

$$D(\lambda, \lambda_0) = D_{589}(\lambda) - D_{589}(\lambda_0) \quad (6)$$

Where λ is the wavelength, λ_0 is the focus wavelength which is 555 nm, and D is the lateral chromatic aberration represented in microns.

Appendix C: Image Quality Metrics

The image convolution metric computes the correlation between the source and retinal image by finding the maximum value of their convolution.

$$\text{Correlation} = \max(A * B) \quad (7)$$

where A is the source image and B is the retinal image.

The Visual Strehl Ratio is an inner product of the PSF with a neural weighting function normalized to the diffraction-limited case. The first equation is the normalized result. The second equation is the neural weighting function as a function of spatial frequency, denoted by variable f.

$$VSX = \frac{\int_{psf} PSF(x,y)\, N(x,y) dxdy}{\int_{psf} PSF_{DL}(x,y)\, N(x,y) dxdy} \quad (8)$$

$$N(f) = 2.6 * (0.0192 + 0.114 * f) * \exp(-(0.114 * f)^{1.1}) \quad (9)$$

The constants provided in equation 9 were obtained from the Matkovic paper that describe the eye model expectation and standard deviation for the Gaussian function.

Appendix D: Polychromatic PSF

Color images were produced by combining three red, green, and blue weighted PSFs based on the wavelengths and weights below. The spectral sensitivity was based off of the data from the 1931 color matching system. Given in XYZ color space, this was converted to RGB through the sRGB working space matrix and then used to weight the PSFs.

```
% weight based on spectral efficiency of the eye
sensxyz = csvread('spectralresponse.csv',1,1);
sensrgb = zeros(length(sensxyz),3);
row = 1;
Msrgb = [3.1338561 -1.6168667 -0.4906146; -0.9787684  1.9161415  0.0334540; 0.0719453 -0.228
for row = 1:length(sensxyz)
    xyz = sensxyz(row:row,:)';
    rgb = Msrgb*xyz;
    sensrgb(row:row,:)=rgb';
    row = row+1;
end
wavelengths = [0.360:0.005:0.830]';
rwvl = [wavelengths sensrgb(1:5:end,1)];
gwvl = [wavelengths sensrgb(1:5:end,2)];
bwvl = [wavelengths sensrgb(1:5:end,3)];
```

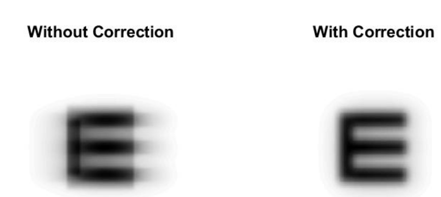
Figure 13: Sample MATLAB code showing our weighting process for for the RGB spectrum.

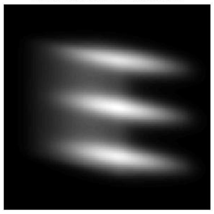
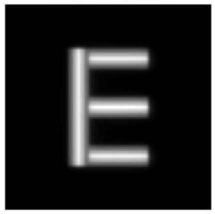
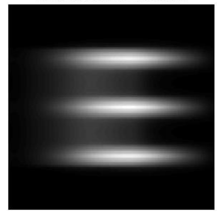
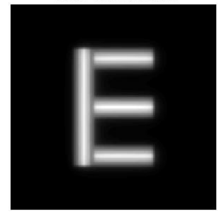
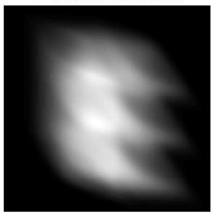
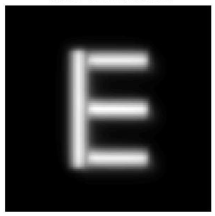
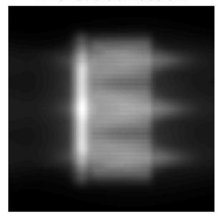
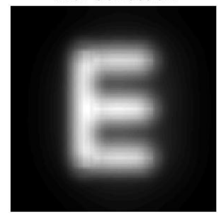
```
% Function that returns a chromatic psf
psf = zeros([n n]);
% Iterates through the spectrum and sums up the psf by the weights of each
% chromatic psf
for i = 1:length(spectrum)
    % Obtains chromatic defocus dependent on wavelength
    chromaticDefocus = chromaticDefocusZernike(spectrum(i, 1), wvl, pupilDiameter);
    zernike_coeffs(2, 3) = chromaticDefocus;

    % Iterate through all the zernike coefficients to obtain a sum of
    % zernikes
    phase = 0;
    for j = 1:length(zernike_coeffs)
        phase = phase + zernike_coeffs(j, 3).* ...
        computeZernike(zernike_coeffs(j, 1), zernike_coeffs(j, 2), rho, theta, true);
    end

    % Obtains the complex pupil function with a circular mask
    pupil = circle.*exp(-1i.*2.*pi./spectrum(i, 1).*phase);

    % sum psf against psf for each wavelength multiplied by each spectrum
    psf = psf + spectrum(i, 2).*abs(fftshift(fft2(fftshift(pupil)))).^2;

end
% Normalize psf
psf = psf ./ max(max(psf));
```

Figure 14: MATLAB code demonstrating the function that computes the chromatic PSF.

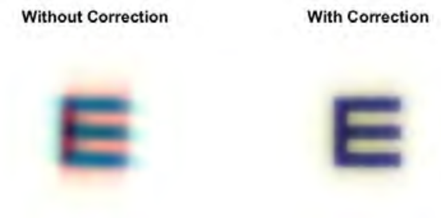
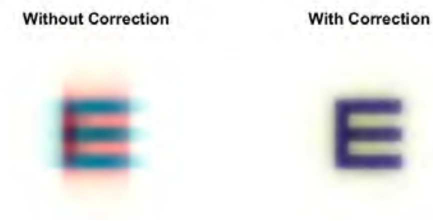
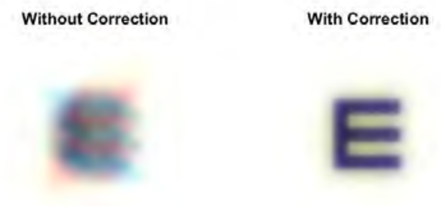
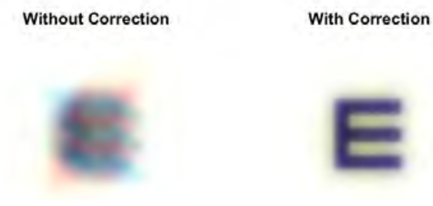
Appendix E: Images Based on Patient Data

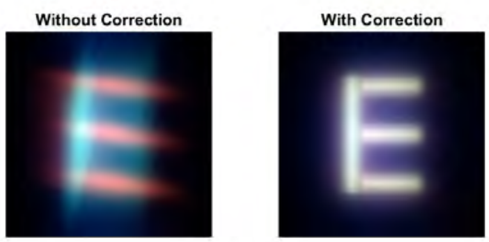
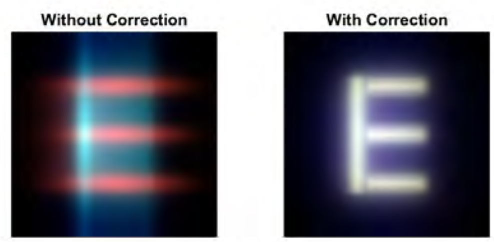
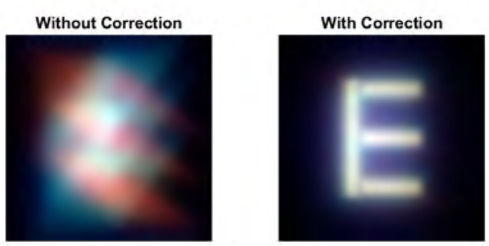
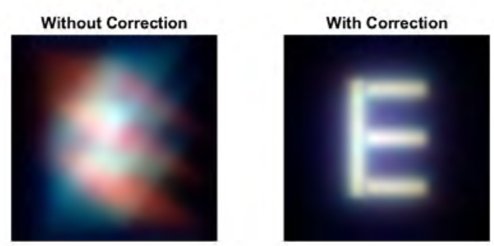
We gathered images from Grayscale.m and Polychromatic.m Below are a compilation of tables that show the vision of patients before and after prescription, each with different images. The first two tables are a continuation of images from Grayscale, as mentioned in the section *Current Patient Data provided by DigitEyez.* The final three tables are of the results from Polychromatic.m using the provided patient Zernike coefficients, each with one of the three images used.

Patient A	**Patient B**
	
Patient C	**Patient D**
	

Patient A	**Patient B**
Without Correction  With Correction 	Without Correction  With Correction 
Patient C	**Patient D**
Without Correction  With Correction 	Without Correction  With Correction 

Patient A	**Patient B**
Without Correction  With Correction 	Without Correction  With Correction 
Patient C	**Patient D**
Without Correction  With Correction 	Without Correction  With Correction 

Patient A	Patient B
	
Patient C	**Patient D**
	

Patient A	Patient B
	
Patient C	**Patient D**
	

The files are provided in a zip file, the README.txt file specifies how to run each file and what each file does.

Special Thanks:
Brandon Zimmerman - Customer
Jennifer Hunter - Faculty Advisor
Wayne Knox - Senior Design Professor
Len Zheleznyak - Image Simulation Advice
James Fienup - Fourier Optics Advice
Anthony Visconti - Gradient Index Modeling in CODEV