Three-Dimensional Gaze Contingent Scotoma Simulation

Sophie Davidson

IMT Atlantique

Contents

Introduction

A scotoma is a blank or impaired region in the visual field, often caused by eye disease such as age-related macular degeneration [1]. The presence of a scotoma can have a serious impact on a patient's quality of life, limiting their ability to read, drive, or to recognise faces (**Erreur ! Source du renvoi introuvable.**).

The aim of this study is to simulate a gaze-contingent scotoma in 3D space using eye-tracking technology and a 3D display using active shutter glasses. Gaze contingency has been of recent interest due to the related advancements in augmented or virtual reality technology. The potential benefits of this technology are apparent in the field of optics and psychology. Recent studies have been successful in the simulation of a 2D gaze-contingent scotoma [2]. By simulating the scotoma using the 3D display and shutter glasses, it will be possible to consider the effects of asymmetry or relative scotoma position on the overall visual field, extending on the existing research.

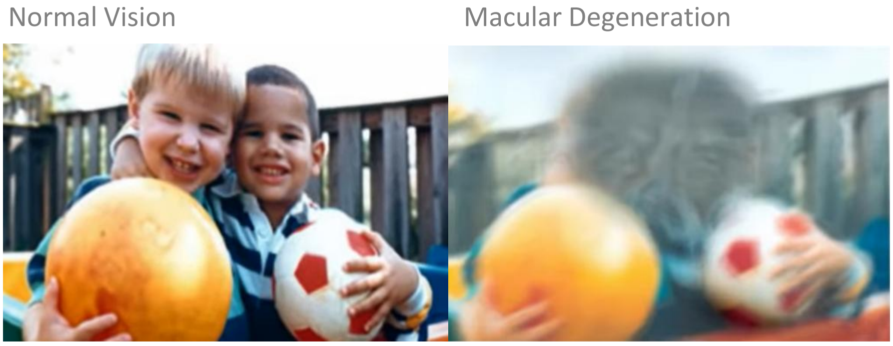


Figure 1: Effect of Macular Degeneration on Vision [17]

Scope

The aim of this study is to develop a system capable of simulating a scotoma in three-dimensional space using eye-tracking technology and LCD shutter lenses. To exploit the three-dimensional nature of the project, both symmetric and asymmetric scotoma should be considered. The produced system should be aligned with the needs and interests of the psychophysics research field. The system should be well documented, and easy to use such that it can be operated without an extensive technical background, and setup time is minimised where possible to create an "out of the box" system.

The scope of this project is limited to the development of new software to compliment and extend the functionality of existing and available hardware and software developed by Pupil Labs and Orthoptica.

Equipment:

Hardware

Pupil Core Eye-Tracker

The Pupil Core eye-tracker (Figure 2) was developed by Pupil Labs as a cost-effective and research ready solution [3]. The eye-tracker is paired with an open-source software suite which allows for real-time viewing, or post-hoc analysis of the eye-tracking data [4]. Several plug-in modules allow for extended functionality including surface tracking, streaming of data through a local network, and blink detection.

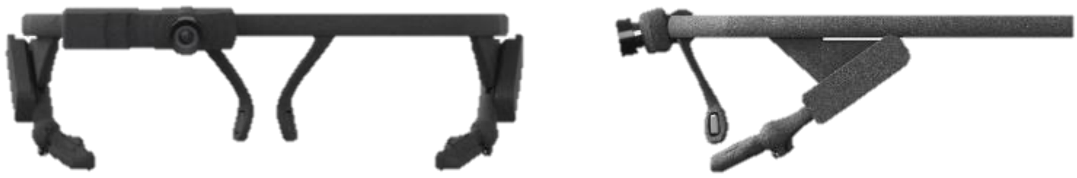


Figure 2: Pupil Core Eye-tracker [3]

The network Application Programming Interface (API) was designed to increase the adaptability and extendibility of the software by streaming data directly from the Pupil Capture program. Data is transmitted through the API using ZMQ, an open-source messaging library using the MessagePack protocol (**Erreur ! Source du renvoi introuvable.**). These platforms are both optimised for high speed and lightweight applications, promoting their use in real-time systems [5] [6]. Python packages for both ZMQ and MessagePack are available, facilitating access to the API, with only some basic configuration required [7] [8].

Communicating with the eye-tracker in this way allows for real-time access to gaze position, surface information, and remote control of the device among other things. By making use of the network API, gaze contingency becomes viable.

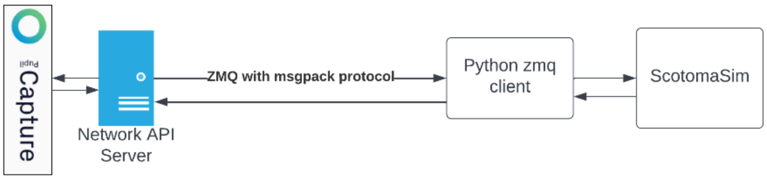


Figure 3: Network Diagram

Active 3D Shutter Glasses

Stereo shutter glasses contain a liquid crystal layer which can be switched rapidly between opaque and translucent, blocking one eye at a time [9]. A display such as a projector can then be synchronised with the shutter glasses such that a different image is seen by each eye (Figure 4) [9]. If the refresh rate is sufficiently high, the wearer cannot differentiate between the two frames, and fuses the two images together, creating the appearance of a three-dimensional image [9]. In the case of this experiment, the ability of this technology to display separate images can be exploited to simulate the effect of eye-disease which affects each eye disproportionately.

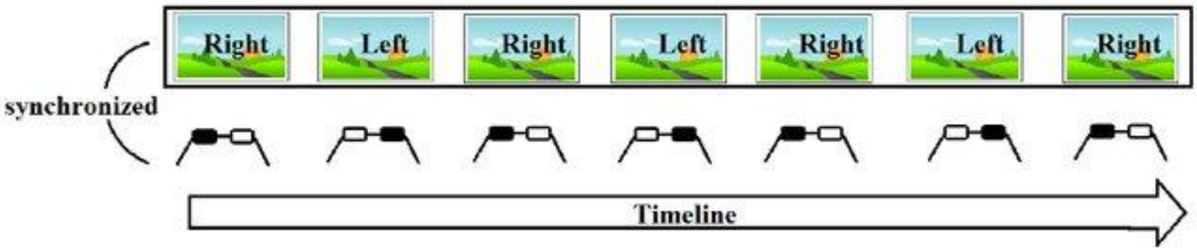


Figure 4: 3D Display Synchronisation for Shutter Glasses [10].

April Tags

April tags (Figure 5) are a type of fiducial marker placed in the field of view to create a reference point for a computer imaging system. The Pupil Capture "surface tracker" uses these markers to define a surface in 3D space. The gaze data is then applied to the defined surface to find the gaze position on that surface [11]. If more than two markers are used, Pupil Capture can estimate the surface even if some markers are obscured or if detection fails [11]. For simulation, four April tags were used to increase the robustness of surface tracking, and to mitigate the effects of poor lighting or environmental changes.

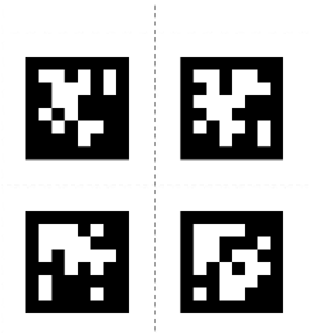


Figure 5: April Tag Markers [11]

Performance:

The stated gaze accuracy of the pupil core eye tracker is 0.60°, with a precision of 0.02° [3]. To determine the experimental performance, a series of tests were completed. The accuracy and precision were measured using the Pupil Capture plug-in, and then recorded and averaged, the results of which can be seen in Table 1. The tests were then repeating using a chin rest to determine if the stabilisation provided would have a significant impact on the resultant accuracy and precision.

Method	Average Accuracy (visual °)	Average Precision (visual °)
Stated Accuracy	0.60	0.02
Without Chin Rest		
Pupil Capture Validation Tool	1.7427 ±0.1287	0.1131±0.0016
With Chin Rest		
Pupil Capture Validation Tool	1.4193±0.1008	0.0841±0.0016

Table 1: Comparison of Precision and Accuracy

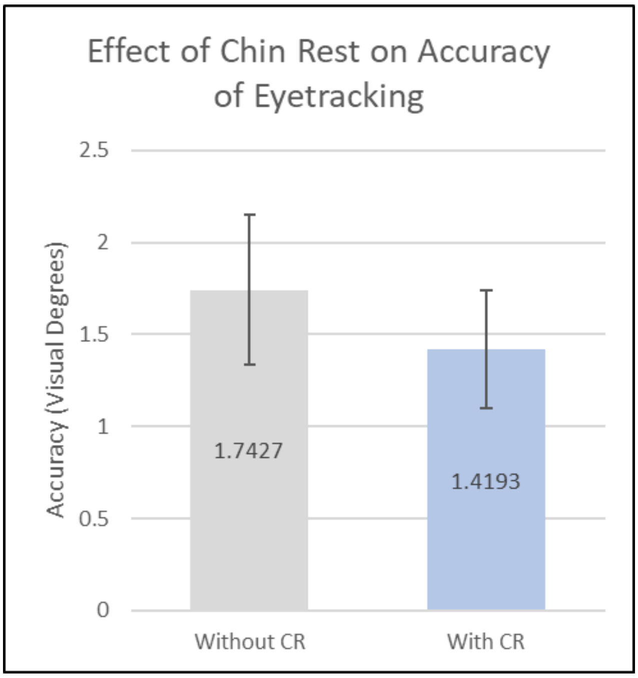


Figure 7: The Effect of Chin Rest on Accuracy with Standard Deviation Bars

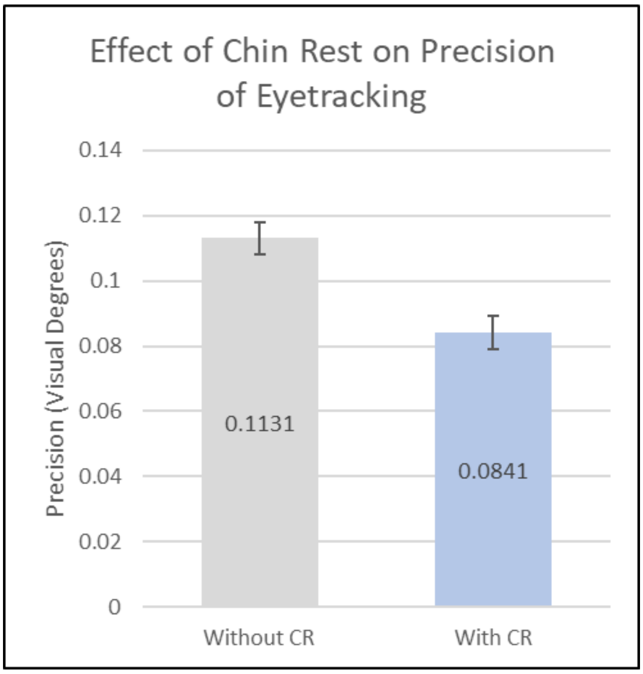


Figure 7: The Effect of Chin Rest on Precision with Standard Deviation Bars

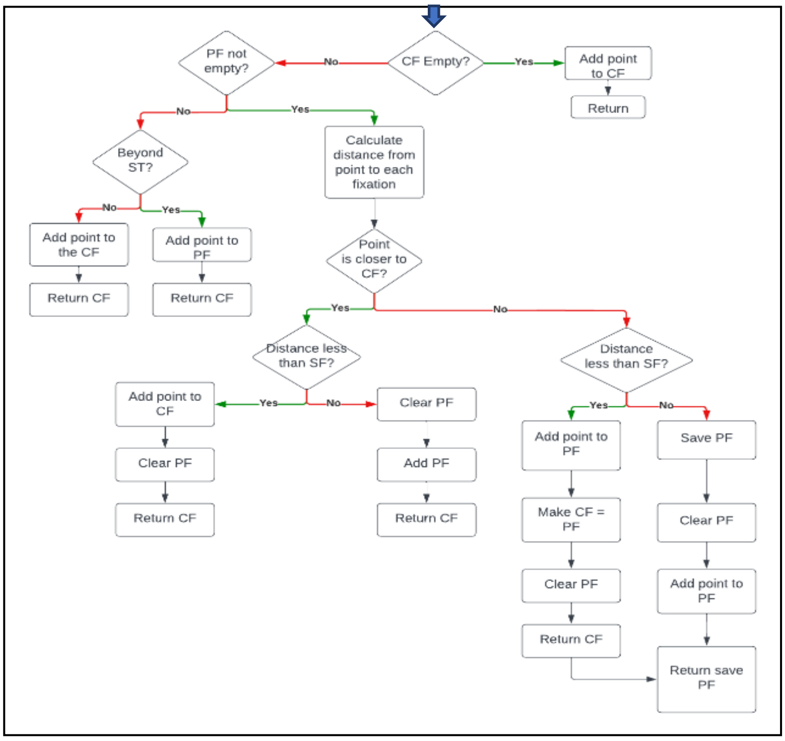
The results shown in Figure 7 demonstrate that there is possibly a significant relationship between accuracy and the use of a chin rest, however there was a large degree of variation between results in accuracy . The effect is more marked on the precision of eye tracking, where there is no overlap in the error bars, and proportionally, a greater difference.

Smoothing

Data received from the eye-tracker can be noisy due to errors in tracking, and due to high-frequency natural eye movements [12]. In an algorithm proposed by M. Kumar(Figure 8), eye-tracker data is analysed to determine whether the gaze data received is part of a saccade, or an abnormal data point [13]. The algorithm looks one sample ahead to determine whether the current point represents the start of a saccade by comparing it to some “saccade threshold.” The resultant fixation is then stored in the current fixation window, which is used to calculate a fixation point by applying a one-sided triangular filter, which favours more recent points (weighted average) to the fixation window (Equation 1).

$$p_{fixation} = \frac{1P_0 + 2p_1 + \cdots + nP_{n-1}}{(1 + 2 + \cdots + n)}$$

Equation 1: One Sided Triangular Filter [13]



CF : Current Fixation

PF : Potential Fixation

ST : Saccade Threshold

Figure 8: Algorithm Flow Chart

Pseudo-code for the gaze smoothing algorithm produced by M. Kumar is provided in Appendix A.

To apply the smoothing algorithm to the eye-tracking data input from the Pupil Core eye tracker in real time, a modified version of the algorithm was implemented using Python.

By randomly selecting a 50-sample selection of consecutive eye-tracker data, the effect of the smoothing algorithm can be visualised.

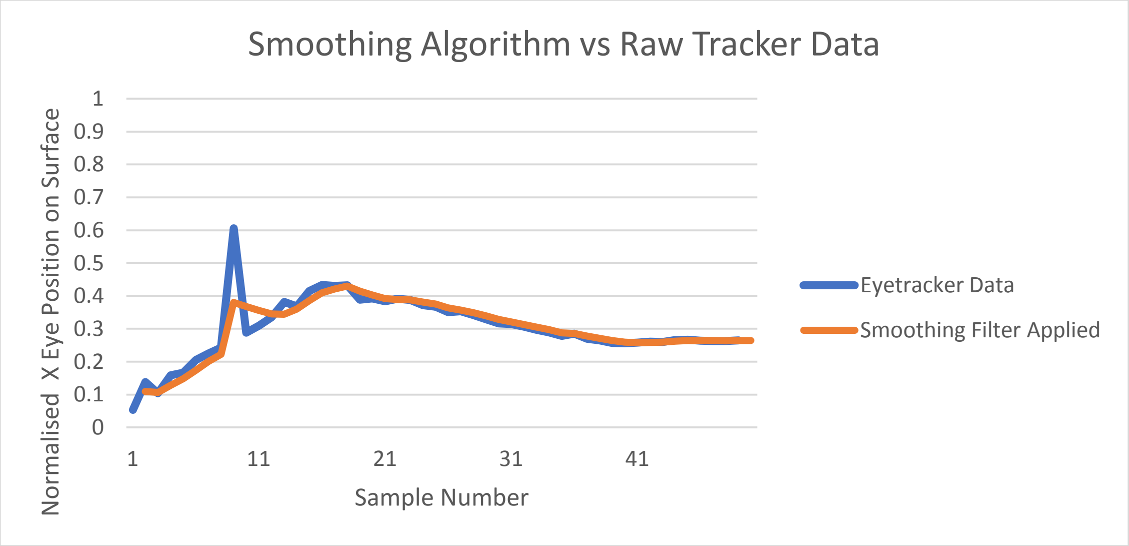


Figure 9: Effect of Smoothing Algorithm

As can be seen in Figure 9, the apparent magnitude of the effect of the saccade occurring around sample 7-10 is reduced. The general appearance of the data is more smoothed, as the algorithm filters the effect of tremors by averaging several samples.

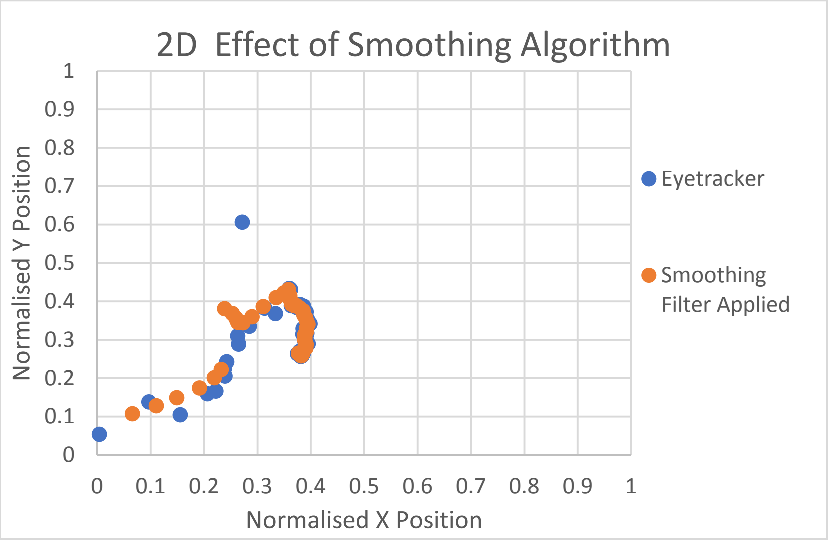


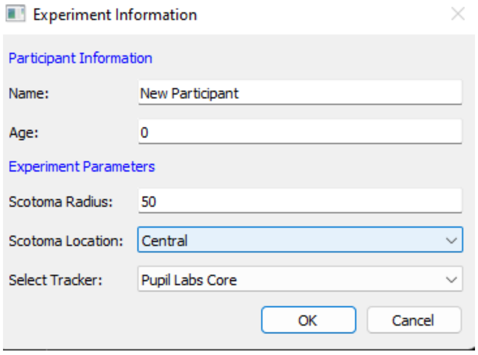
Figure 10: 2D Effect of Smoothing Algorithm

The impact of the smoothing algorithm in 2D can be seen in Figure 10. The saccade can be seen to influence the points which follow it. It may hence be necessary to adjust the saccade threshold and/or the window to achieve a more optimal smoothing effect.

The greater the window size, the more computationally intensive, and hence the slower the program. A balance must be achieved between the required speed, and the efficiency of the smoothing effect. This balance will be dependent on the machine's clock speed, and hence must be variable by the user.

2D Scotoma Simulation

The 2D scotoma simulation system can simulate a symmetric scotoma by using eye position data from the Pupil Core eye-tracker. A circular region is plotted at a relative position from the gaze position. Both the scotoma size and the relative location of the scotoma can be selected by the user (Figure 11).



Participant Information

Name: New Participant

Age: 0

Experiment Parameters

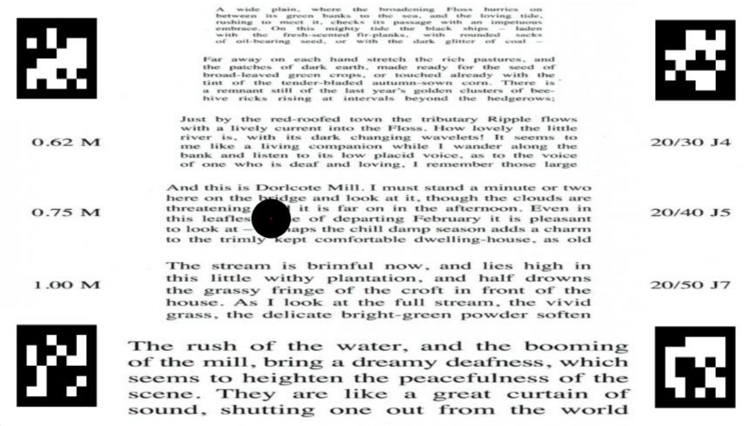
Scotoma Radius: 50

Scotoma Location: Central

Select Tracker: Pupil Labs Core

Figure 11: Experiment Information Dialogue Box

A text background is applied in Figure 12 however, any background is possible depending on the use case of the software. Four April Tags were placed in each corner of the screen such that Pupil Capture software can accurately map the screen without the need for physical tags. At this stage, the multimedia application library *Pygame* was selected due to its relative simplicity and quicker development time.



A wide plain, where the broadening Floss hurries on between its green banks to the sea, and the loving tide, rushing to meet it, checks its passage with an impetuous embrace. On this mighty tide the black ships – laden with the fresh-scented fir-planks, with rounded sacks of oil-bearing seed, or with the dark glitter of coal –

Far away on each hand stretch the rich pastures, and the patches of dark earth, made ready for the seed of broad-leaved green crops, or touched already with the tint of the tender-bladed autumn-sown corn. There is a remnant still of the last year's golden clusters of bee-hive ricks rising at intervals beyond the hedgerows;

Just by the red-roofed town the tributary Ripple flows with a lively current into the Floss. How lovely the little river is, with its dark changing wavelets! It seems to me like a living companion while I wander along the bank and listen to its low placid voice, as to the voice of one who is deaf and loving. I remember those large

And this is Dorlcote Mill. I must stand a minute or two here on the bridge and look at it, though the clouds are threatening it is far on in the afternoon. Even in this leafles e of departing February it is pleasant to look at – aps the chill damp season adds a charm to the trimly kept comfortable dwelling-house, as old

The stream is brimful now, and lies high in this little withy plantation, and half drowns the grassy fringe of the croft in front of the house. As I look at the full stream, the vivid grass, the delicate bright-green powder soften

The rush of the water, and the booming of the mill, bring a dreamy deafness, which seems to heighten the peacefulness of the scene. They are like a great curtain of sound, shutting one out from the world

Figure 12: 2D Scotoma Simulation

By applying the smoothing algorithm to the program, the movement of the scotoma appears more natural to the user.

Stereoscopic Scotoma Simulation

By adapting the 2-D scotoma simulation to a stereoscopic model, it is possible to model the effects of an asymmetrical scotoma and the impact of the generated scotoma on abilities such as depth perception or reading.

The python windowing library *Pyglet* was selected, replacing *Pygame*, for the implementation of the 3D system due to its direct interface with OpenGL, and its multi-window capabilities [14].

OpenGL is a cross-platform API used for rendering graphics, specifically where interaction with a GPU is needed such as for stereoscopic 3D [15]. By offloading to the GPU, performance is accelerated, and details around the configuration can be abstracted away.

To achieve stereoscopic 3D, a NVIDIA Quatro card was required. In the case of this report, a device with both Windows 10 (required for Pupil Capture Software) and the required graphics card was not available, hence the network API was accessed remotely using the device's IP address. In order to minimise latency, a direct Ethernet connection was implemented between the two devices, although wireless connection over a network is also possible. The system described above is visualised in the diagram below (Figure 13).

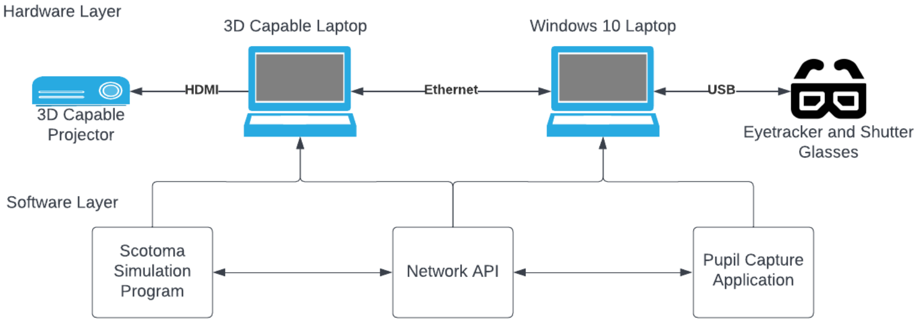


Figure 13: System Diagram

Limitations of the Stereoscopic Model

As the effective refresh rate is halved due to the nature of the stereoscopic display (Table 2), it was determined that the smoothing algorithm should be discarded when in stereoscopic mode. It would be possible, using a system with a higher refresh rate (ie – 240Hz+) to implement such an algorithm with a lesser impact on the visual performance. It is commonly stated that the human eye detects changes up to around 60Hz [16], with only artefacts existing at higher frequencies. Using the existing system, in 3D mode, the output is at just 60Hz per eye, meaning that any slowing of the refresh rate may be detected by the eye. In order to mitigate the effect of saccades, and calibration errors, the eye-tracker should be calibrated to the greatest degree of accuracy and precision possible.

Specification	Value
Projector Refresh Rate	120Hz
Eye-Tracker Refresh Rate	120Hz
Individual Eye Refresh Rate	60Hz

Table 2: Equipment Refresh Rate

Testing

To test the validity of the generated simulation, a test was created. Participants are to read a text from the centre region of the screen. The parameters of the scotoma such as its size, relative location, and offset were adjusted. The simulated scotoma is asymmetric meaning that each side can be adjusted independently of the other using the control panel (Figure 14).

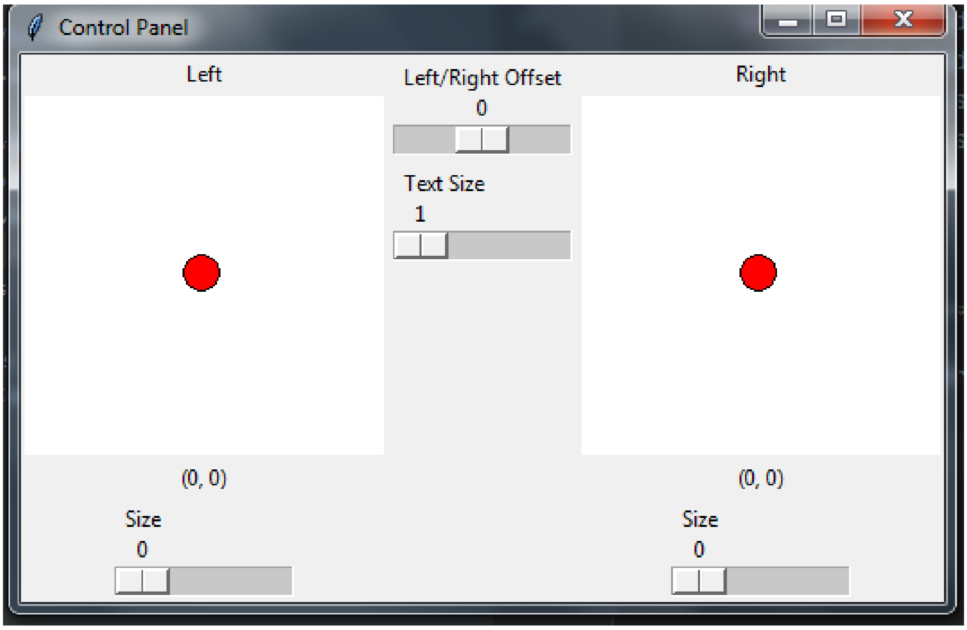


Figure 14: Control Panel Window

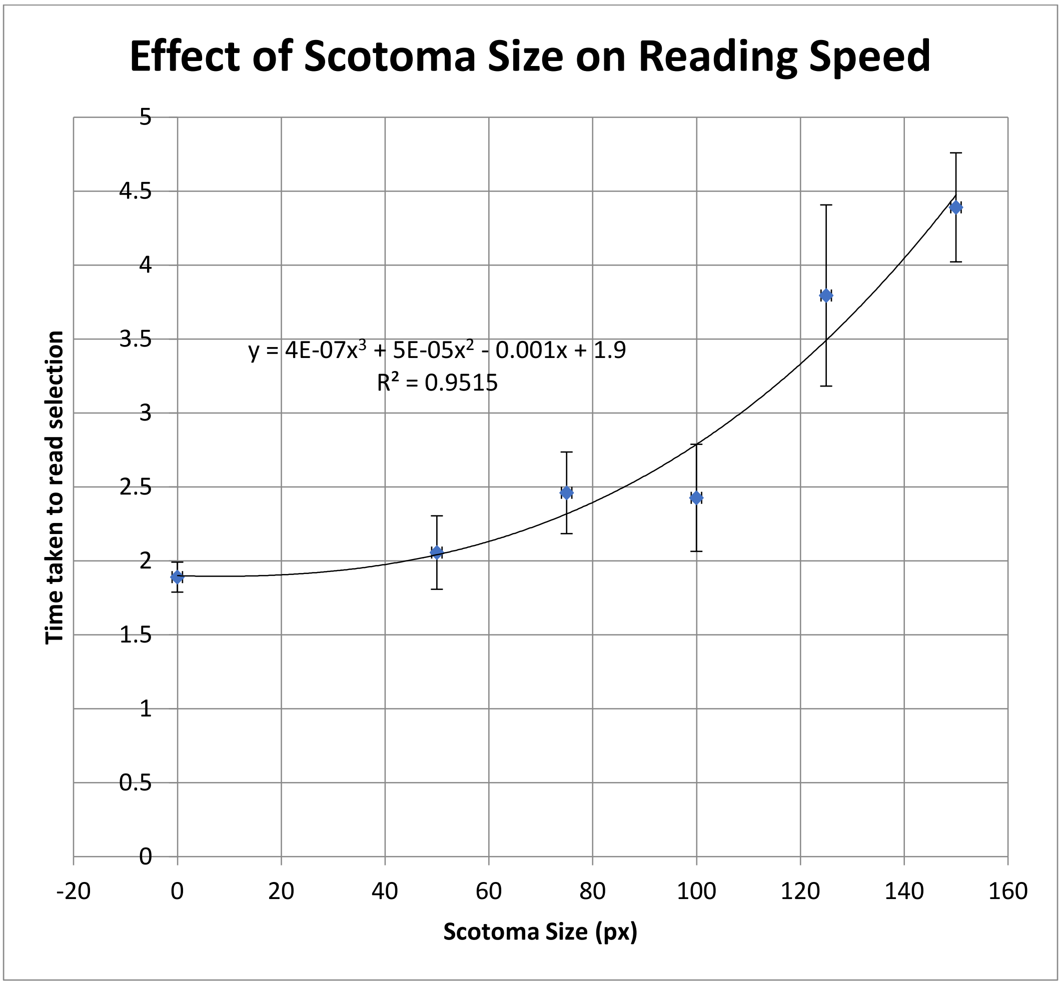
Initial findings were completed using one participant only, for a more comprehensive study, a larger sample size of participants should be used.

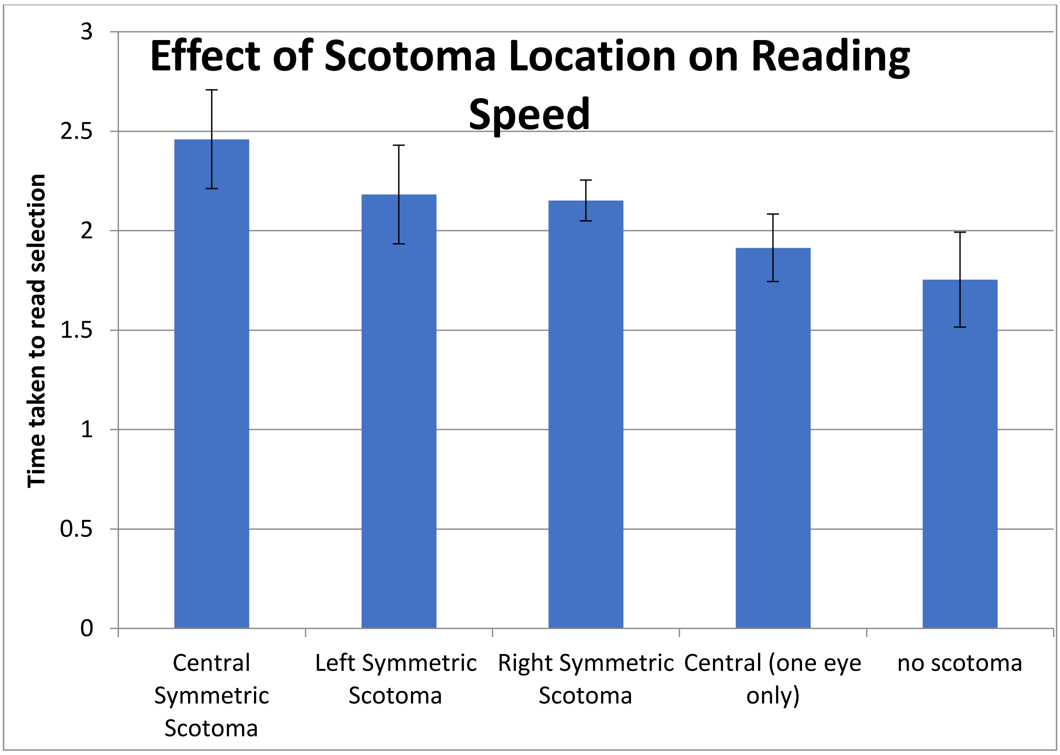
Method:

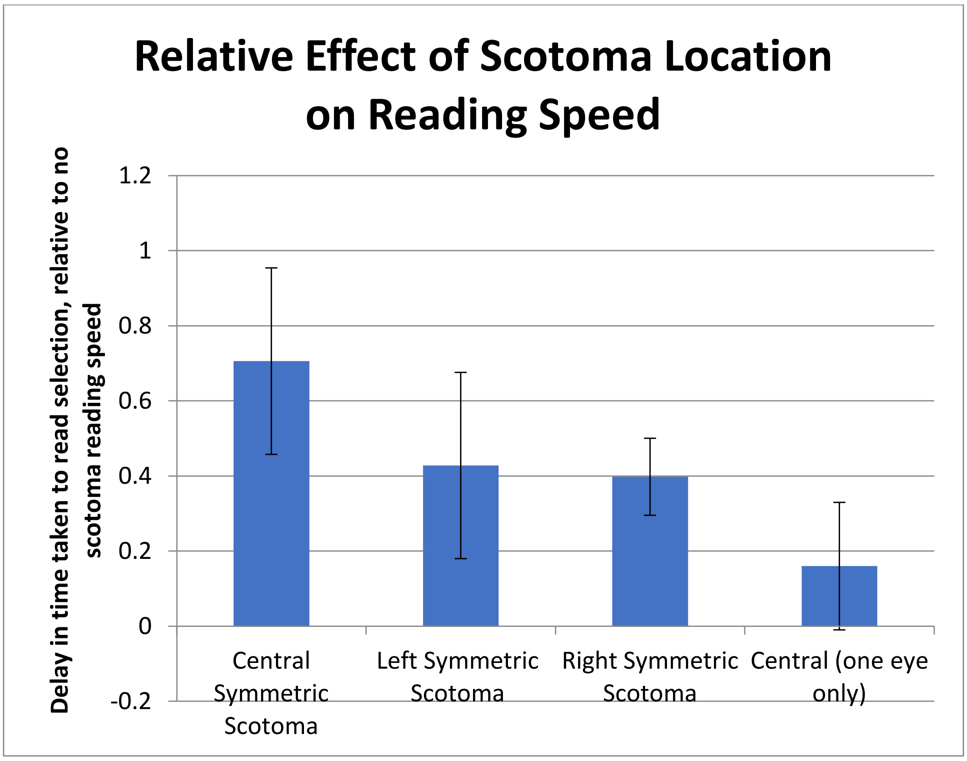
1. A large text was split into sections of 5 words
2. The order of the sections was randomised using the excel RAND() method.
3. The eye tracker was calibrated, ensuring an accuracy of below 2 visual degrees.
4. A selection of text was then displayed on the screen, and the parameters were adjusted using the control panel (text size = 40, scotoma size = 75, offset = 15, position =central)
5. The test was repeated 5 times, using different prompt texts each time, and maintaining the same parameters
6. The test was then repeated, changing the parameters to match the modes below:
    a. (text size = 40, scotoma size = 75, offset = 0, position = (x=-20))
    b. (text size = 40, scotoma size = 75, offset = 0, position = (x=20))
    c. (text size = 40, scotoma size = 75, offset = 0, position = central, left eye off)
    d. (text size = 40, scotoma size = 0)
    e. (text size = 40, scotoma size = 50, offset = 0, position = central)
    f. (text size = 40, scotoma size = 75, offset = 0, position = central)
    g. (text size = 40, scotoma size = 100, offset = 0, position = central)
    h. (text size = 40, scotoma size = 125, offset = 0, position = central)
    i. (text size = 40, scotoma size = 150, offset = 0, position = central)
    j. (text size = 40, scotoma size = 75, offset = 10, position = central)
    k. (text size = 40, scotoma size = 125, offset = 20, position = central)
    l. (text size = 40, scotoma size = 125, offset = 30, position = central)
    m. (text size = 40, scotoma size = 125, offset = 40, position = central)
7. The average, and standard deviation was then calculated for each test.
8. The results were graphed in groups relating to the changed parameter.

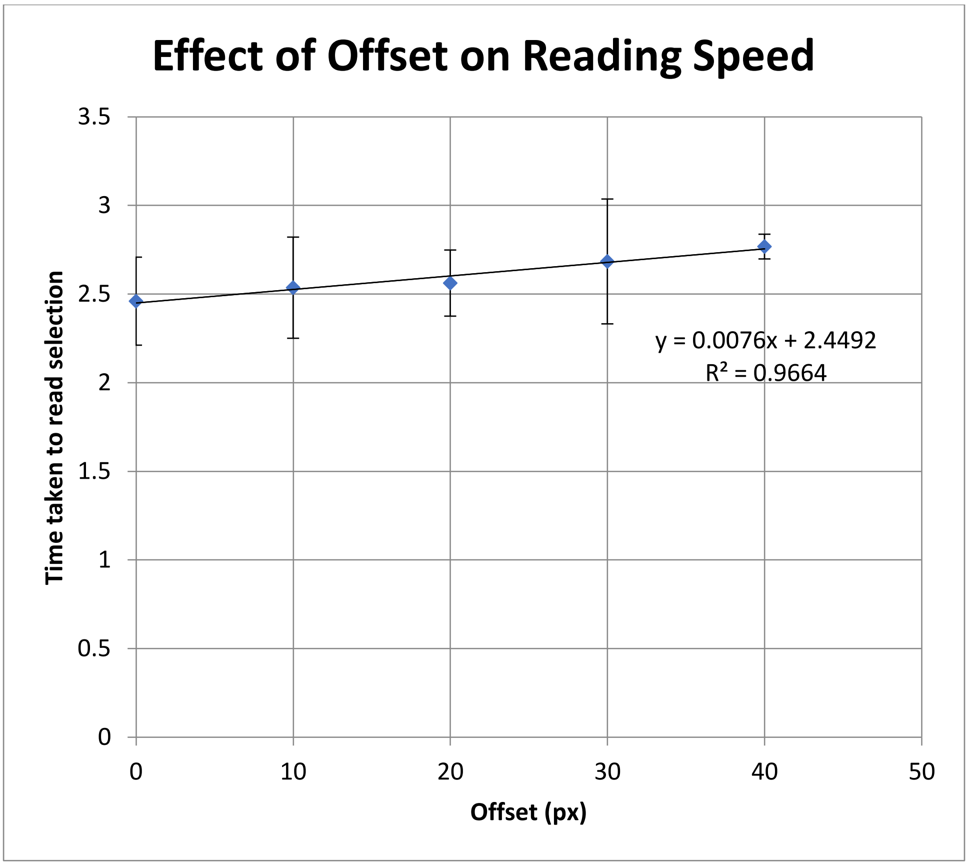
Results

Table of results can be found in Appendix C.




Effect of Scotoma Location on Reading Speed
Time taken to read selection
3
2.5
2
1.5
1
0.5
0
Central Symmetric Scotoma
Left Symmetric Scotoma
Right Symmetric Scotoma
Central (one eye only)
no scotoma


Relative Effect of Scotoma Location on Reading Speed
Delay in time taken to read selection, relative to no scotoma reading speed
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
Central Symmetric Scotoma
Left Symmetric Scotoma
Right Symmetric Scotoma
Central (one eye only)



Based on the above results, it appears that scotoma location and size have the greatest impact on reading speed. The central symmetric scotoma had the greatest perturbing effect on the user, while a central scotoma present in only one eye had very little effect. Scotoma located to the left and right of the user's field of vision had a lesser effect, but still slowed reading speed significantly. It appears that offset (or the 3D "plane") between the scotoma locations in the eyes had very little effect.

Relating to the size of the scotoma, it is apparent that once the scotoma exceeds more than half of the average word size, a greater impact on reading speed was present;

Conclusion

The simulation created for this project is successful in demonstrating the effects of a asymmetric scotoma, and may help us to understand how different eye conditions effect the general field of vision. A possible use of this software could be to generate data on eye-movement patterns in healthy individuals when a scotoma is introduced. This data could then form the training set for a machine learning algorithm which could then be used as a diagnostic tool. The software may also be useful as a training tool in order to assist those with degenerative eye disease to learn adaptive strategies for reading.

Bibliography

[1] Harvard Medical School, “Medical Dictionary of Health Terms,” 13 December 2011. [Online]. Available: https://www.health.harvard.edu/a-through-c#A-terms. [Accessed 15 December 2022].

[2] E. C. Carlos Aguilar, “Evaluatuon of a gaze-controlled vision enhancement system for reading in visually impaired people,” *PLOS One,* pp. 1-24, 2017.

[3] Pupil Labs, “Technical Specs and Performance,” Pupil Labs, [Online]. Available: https://pupil-labs.com/products/core/tech-specs/. [Accessed 15 12 2022].

[4] Pupil Labs, “Pupil Core,” Pupil Labs, [Online]. Available: https://pupil-labs.com/products/core/. [Accessed 15 12 2022].

[5] MessagePack, “MessagePack,” [Online]. Available: https://msgpack.org/. [Accessed 16 12 2022].

[6] ZeroMQ, “ZeroMQ: An open-source universal messaging library,” [Online]. Available: https://zeromq.org/. [Accessed 16 12 2022].

[7] PyPi, “msgpack1.0.4,” 3 6 2022. [Online]. Available: https://pypi.org/project/msgpack/. [Accessed 16 12 2022].

[8] Pypi, “pyzmq24.0.1,” 21 9 2022. [Online]. Available: https://pypi.org/project/pyzmq/. [Accessed 16 12 2022].

[9] H. J. Y. S. Y. W. M. G. Y. C. Y. C. Huang Wu, “Evaluating Sterioacuity with 3D Shutter Glasses Technology,” *BMC Opthamology,* vol. 16, no. 45, 2016.

[10] H. Sarbolandi, “Simultaneous 2D and 3D Video Rendering,” Tampere University of Technology, Tampere, Finland, 2013.

[11] Pupil Labs, “Surface Tracking,” [Online]. Available: https://docs.pupil-labs.com/core/software/pupil-capture/#surface-tracking. [Accessed 16 12 2022].

[12] J. K. T. W. A. P. R. P. Manu Kumar, “Improving the Accuracy of Gaze Input for Interaction,” in *ETRA*, 2008.

[13] M. Kumar, “GUIDe Saccade Detection and Smoothing Algorithm,” Stanford University, Department of Computer Science, California, 2007.

[14] Pyglet, “The OpenGL Interface,” 2020. [Online]. Available: https://pyglet.readthedocs.io/en/latest/programming_guide/gl.html. [Accessed 9 1 2023].

[15] OpenGL, “OpenGL Overview,” Khronos Group, [Online]. Available: https://www.khronos.org/opengl/. [Accessed 1 9 2023].

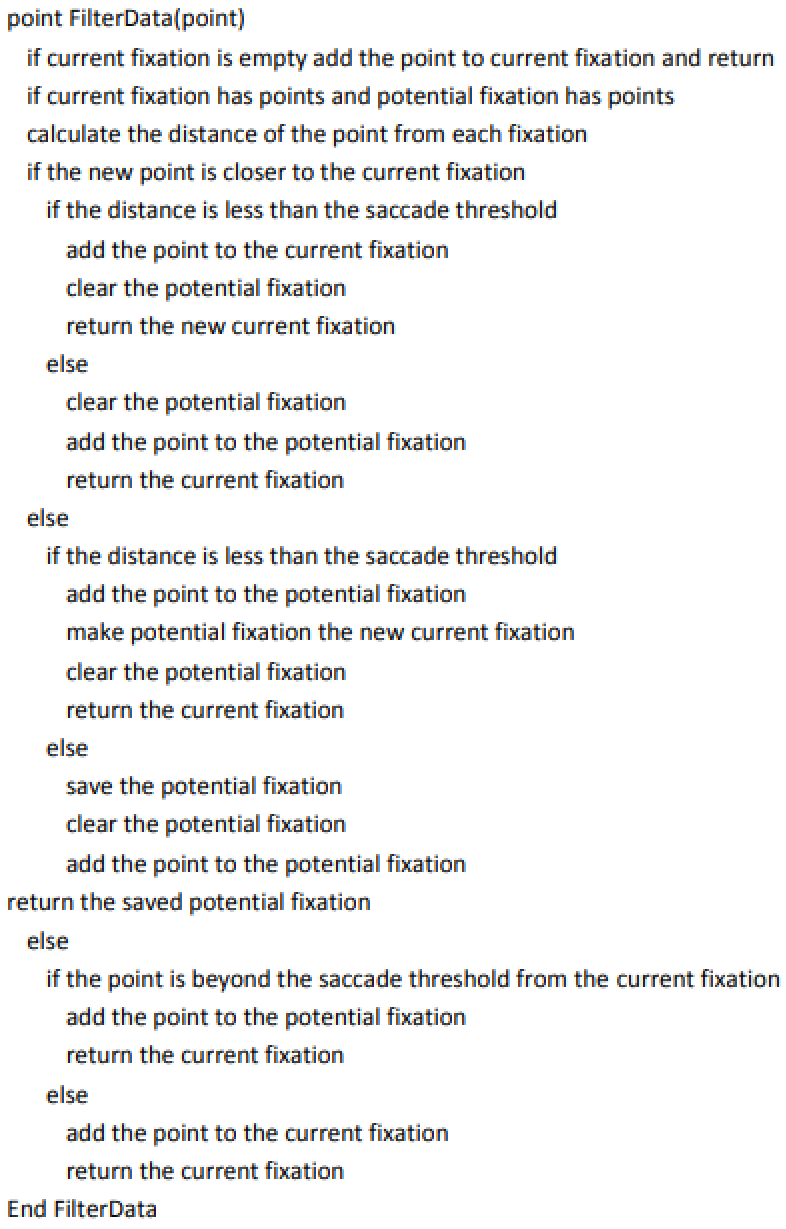
[16] R. S. Randolph Blake, Perception, New York: McGraw- Hill , 2005.

[17] National Eye Institute, “Learn About Age-Related Macular Degeneration (AMD),” National Institute of Health, 2011.

[18] SHARP/NEC, “3100-lumen Widescreen Ultra Short Throw Projector,” SHARP, [Online]. Available: https://www.sharpnecdisplays.us/products/projectors/np-u310w. [Accessed 23 12 2022].

[19] Open GL, “Getting Started,” 2022. [Online]. Available: https://www.khronos.org/opengl/wiki/Getting_Started. [Accessed 2023 1 9].

Appendix A – Pseudo-Code for Gaze Smoothing Algorithm



```
point FilterData(point)
  if current fixation is empty add the point to current fixation and return
  if current fixation has points and potential fixation has points
  calculate the distance of the point from each fixation
  if the new point is closer to the current fixation
    if the distance is less than the saccade threshold
      add the point to the current fixation
      clear the potential fixation
      return the new current fixation
    else
      clear the potential fixation
      add the point to the potential fixation
      return the current fixation
  else
    if the distance is less than the saccade threshold
      add the point to the potential fixation
      make potential fixation the new current fixation
      clear the potential fixation
      return the current fixation
    else
      save the potential fixation
      clear the potential fixation
      add the point to the potential fixation
return the saved potential fixation
  else
    if the point is beyond the saccade threshold from the current fixation
      add the point to the potential fixation
      return the current fixation
    else
      add the point to the current fixation
      return the current fixation
End FilterData
```

Figure 15: Algorithm Pseudocode [13]

Appendix B - Eye-tracking Performance Measurement

Accuracy

Trial Method	1	2	3	4	5	6	7	8	9	10	Average	Std. Devn
Without Chin Rest												
Pupil Capture Validation	1.845	1.304	1.587	1.425	1.172	1.489	2.437	2.385	1.878	1.905	1.7427	0.407113
With Chin Rest												
Pupil Capture Validation	1.876	1.96	1.748	1.343	1.211	1.504	1.175	1.269	1.078	1.029	1.4193	0.318818

Precision

Trial Method	1	2	3	4	5	6	7	8	9	10	Average	Std. Devn
Without Chin Rest												
Pupil Capture Validation	0.113	0.118	0.116	0.103	0.112	0.107	0.112	0.121	0.117	0.112	0.1131	0.005029
With Chin Rest												
Pupil Capture Validation	0.085	0.073	0.078	0.082	0.084	0.091	0.092	0.084	0.085	0.087	0.0841	0.005089

Appendix C – Testing Results

	Time to read selection (s)						
Scotoma Location	1	2	3	4	5	Average	STDEV
Central Symmetric Scotoma	2.2	2.25	2.62	2.44	2.79	2.46	0.248294
Left Symmetric Scotoma	2.29	2.47	2.25	2.09	1.81	2.182	0.248032
Right Symmetric Scotoma	2.05	2.09	2.23	2.1	2.29	2.152	0.102567
Central (one eye only)	1.77	1.72	2.11	1.92	2.05	1.914	0.169794
no scotoma	1.85	1.43	1.58	1.92	1.99	1.754	0.2386
Stotoma Size (0 offset)							
50px symmetric	2.09	2.18	1.91	2.09	2.01	2.056	0.10139
75px symmetric	2.2	2.25	2.62	2.44	2.79	2.46	0.248294
100px symmetric	2.75	2.69	2.22	2.15	2.32	2.426	0.275917
125px symmetric	4.19	3.69	3.8	4.04	3.25	3.794	0.361981
150px symmetric	3.89	4.4	4.57	3.78	5.31	4.39	0.612577
0px scotoma (none)	2.2	1.43	1.58	2.25	1.99	1.89	0.368578
Offset (symmetric only)							
0px offset	2.2	2.25	2.62	2.44	2.79	2.46	0.248294
10PX	2.06	2.65	2.82	2.61	2.54	2.536	0.285359
20PX	2.46	2.85	2.41	2.44	2.65	2.562	0.186467
30PX	2.85	2.08	2.85	2.96	2.68	2.684	0.352179
40PX	2.74	2.78	2.79	2.86	2.67	2.768	0.069785