Summer Scholarship Summary Report

Researchers: Peter Mitchell, Dr. Brett Wilkinson, Dr. Sean Fitzgibbon

Aim

The aim of this summer scholarship was to develop and test a stimulus system to deliver steady state visual stimuli on a computer screen for use in Brain Computer Interface (BCI) research. The key design criteria were:

1. The stimulus would elicit a steady-state visual evoked potential (SSVEP) in the user
2. The stimulus could be overlayed on video playback or on a computer game
3. The flashing rate of the stimulus should be configurable
4. The size of the stimulus should be configurable

Ideally the apparatus would be able to present a small image flashing at a high frequency overlayed on a video playback background and still be able to elicit an SSVEP in the user.

Project Summary

The work completed as part of this summer scholarship primarily included the following activities:

- Development of applications using XNA with C# and DirectX with C++ to act as prototypes in piloting parameters for use in SSVEP.
- Completion and analysis of a series of equipment tests to ensure that the system meets the basic requirement to elicit an SSVEP in the user, and to pilot parameters suitable for use of the system in research experiments

The initial application was developed in C# with XNA due to the immediate familiarity from work with other recent projects. The XNA system would have allowed for implementation with simple games developed with rapid prototyping potentially in future tests. At first due to a misunderstanding of the requirements and the way SSVEP worked the application was developed to desynchronise the on screen rendering by splitting the screen into a grid. This was abandoned as it did not in itself provide the stimulus that would be required for an SSVEP application. The initial application provided a set of simple options allowing changing between a variety of backgrounds (in particular: video, black, and white), the colour of the stimulus and other settings as described on the next page.

The limitations of XNA over how the graphics and communication with the graphics card can be controlled led to a language and library swap to DirectX using C++. Many games are commonly developed with C++ and DirectX so it was logical to develop prototype code in a more transferable environment. No further updates were made to the XNA with C# build, but by the end of development the C++ version of the application provided a set of 56 controls. These 56 controls (described later in the document) could be summarised into 31 variables. These could be stored into a save file and up to 20 different combinations of options could be configured and rapidly toggled between. This feature provided an ease of use for testing as the original version of the application did not support pre-configuration of test parameters.

Summary of XNA C# Application Features

Figure 1: XNA and C# Prototype

Figure 1 shows an example screenshot of the XNA and C# application with the sample video playing and the control panel located in the top left region of the window. On the following page is a short summary of the features provided by this application.

The XNA and C# version of the application was developed using a custom GUI overlay the controls this overlay provides access to within the application were as follows:

- The option to hide and show the controls. This was included to make it possible to limit the amount of space consumed on screen by the overlay.
- The option to show 4 different backgrounds to any of:
    - a plain white background.
    - a plain black background.
    - a background that continually changes colour between a red and a blue background.
    - a background that runs as a video showing a short video.
- The option to change the stimulus (flash colour) to any of:
    - black.
    - white.
    - red.
    - green.
    - blue.

- The option to change the rate of flashing to any value including:
  - 1, 2, 3, or 4 Hz.
  - 5 Hz and then every increment of 5Hz up to and including 100Hz.
- The option to modify how the stimulus is rendered as any of:
  - using the “Graphics Clear” method.
  - using a rendered texture.
  - using a rendered texture and splitting the screen into a grid. The grid side was controlled using the options for Grid W (width) and Grid H (height). The screen would be divided into this many regions in each direction and only the region with the mouse currently over it would flash.
- The option to enable or disable Vertical Retrace Synchronisation. This option controls whether the game loop waits for the monitor’s refresh cycle to end before rendering the next frame. When enabled this option prevents tearing, but it limits the frame rate to the refresh rate of the monitor.
- The option to set a preferred Frames Per Second (FPS) that should be rendered. This value would not change anything when the Vertical Retrace Synchronisation is enabled. It would however allow restriction of the redraw rate of the application. Values allowed started at 50 FPS and went up to inclusive of 300 FPS in 50 FPS increments.
- The actual FPS and the time per frame (in ms) was displayed on the top right of the screen.
- Finally the option to save and load the configured settings was included. The option in this version of the application only allowed for one saved configuration.

In addition to these features the application hidden in the code provides another alternate way of performing the rendering and other specialised activities. The application when using the code would split the screen into defined regions. These regions could then have their properties individually controlled. The cells would update based on a “cooldown” mechanism to change the rate at which they were being rendered to the screen. This method was abandoned as it did not meet requirements for providing stimulus.

Summary of DirectX C++ Application Features

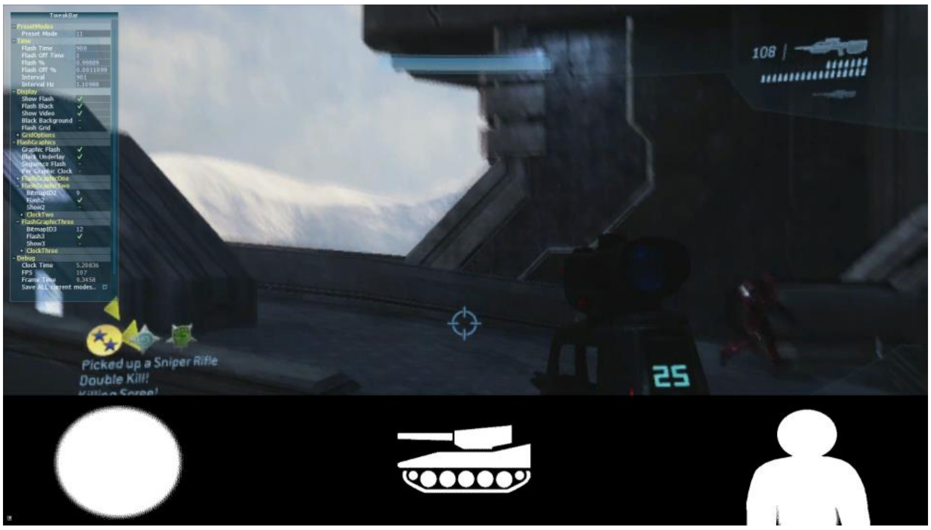

Figure 2: DirectX and C++ application showing example graphics that could be flashed

For a full list of all the functionality of this application please see the ReadMe file that is with the application. The full details are too extensive for this report and so a simple summary of the types of functionality that are provided is here. The application was redeveloped using C++ and DirectX to provide a more controllable environment to work in. The GUI system used is AntTweakBar. This library provides a control panel that options can be added to for manipulating elements of the application. This was perfect for this type of simple development. The main features of the application include:

- Full screen or grid based flashing at a specified rate. The rate can be controlled by setting a time the flash is visible and how long then is isn't before the next flash. The grid based flashing could be controlled by using the position of the mouse, a specified grid coordinate on the screen or other cycling flashing such as an edge flash that would move in sequence around the left and bottom edges of the screen. The grid size could also be specified.
- A specialised additional system was incorporated to provide flashing of images rather than a plain colour. The images (3 places at the bottom of the screen) could have any of them enabled or disabled. The flashing could be set to all use the same flash on and off times, the could have a sequence of flashing much like the edge flashing or they could be set to have individually controlled flash times. This would allow the three to potentially be distinguishable should further testing go on to use this feature. Figure 2 shows an example of some of the included graphics. These were not used in any of the completed tests.

The majority of options were not required for the experiments. Each experiment has detailed the specific settings that were used.

Equipment Tests

Equipment tests were conducted to ensure that the system meets the basic requirement to elicit an SSVEP in the user, and to pilot parameters suitable for use of the system in research experiments. These tests were as steps in the iterative development process of the system to ensure that the system was operating as intended. Therefore the results must be interpreted in this context and, whilst promising, the results are inconclusive without proper evaluation in a controlled experiment.

First Equipment Test: Colour

The purpose of the first test was to:

- Confirm that the whole screen flashing elicits an SSVEP in the user
- Pilot white and black stimulus overlays on contrasting background (i.e. black on white, white on black) and on video playback background

Figure 3: XNA with C# application

The first test was to ensure the viability of using the BCI equipment to recognise the input for SSVEP. An application was developed using XNA and C# as can be seen above in Figure 3. The panel on the upper left provided the custom controls that could be used to modify the visual state of the application. Refer to the earlier section in this document covering the XNA and C# application for additional details. The tests completed were:

- Full screen white stimulus on black flashing at 15Hz
- Full screen black stimulus on white flashing at 15Hz
- Full screen white stimulus on video flashing at 15Hz
- Full screen black stimulus on video flashing at 15Hz

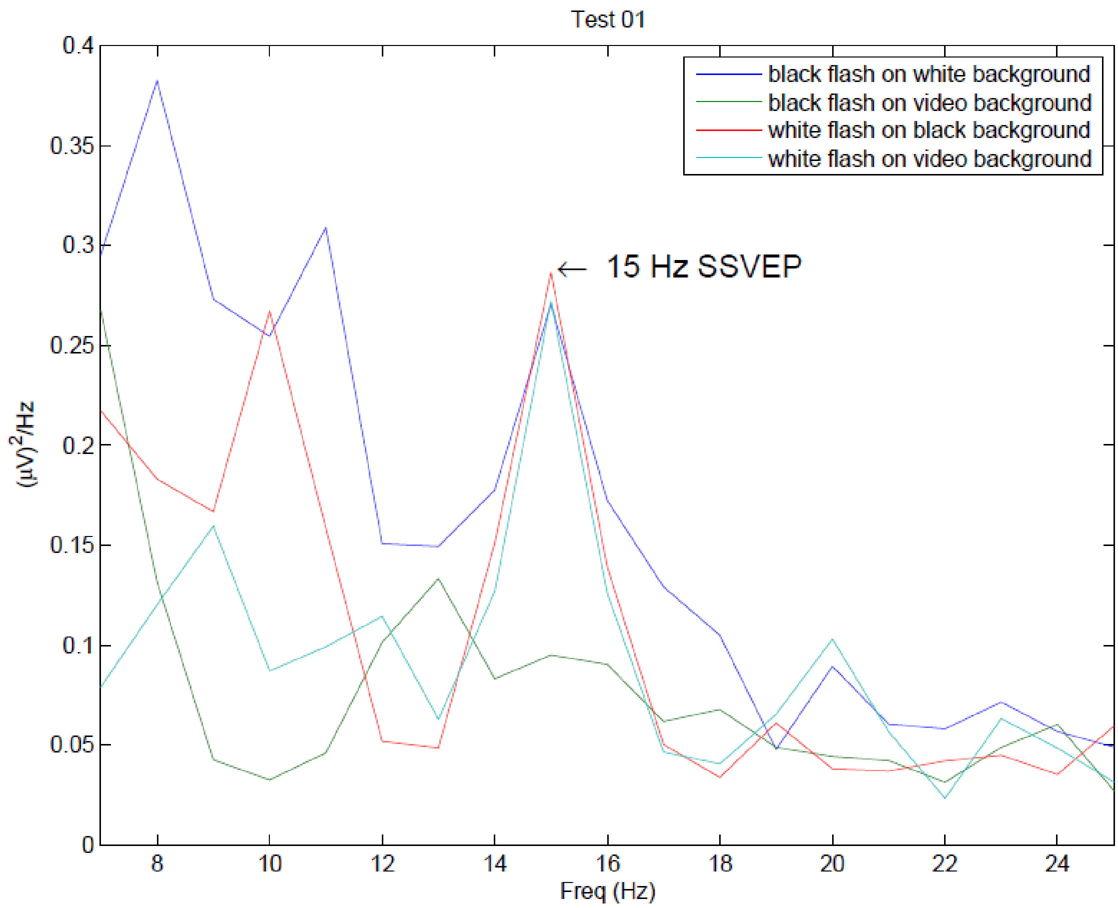

Figure 4: First Equipment Test Results (Colour)

Figure 4 shows results from one participant for this first equipment test. The distinct spike at 15Hz demonstrates a high reaction to the stimulus for the black on white, white on black, and white on video. The black on video resulted in no real spike at 15Hz. It is suspected this is because the video was dark in appearance as seen in many of the earlier figures (eg, Figure 3) and so the lack of contrast reduced the effectiveness as a stimulus.

This test demonstrated that the high contrast between a flat white and the darkness of the video or a black was important for SSVEP. This suggests that if this method of interaction were to be used within an interactive game context the contrast would need to be considered as part of the design. The full screen flash used in this test would probably not be useful in real scenarios as the long term intention is to identify when an individual is observing a region of the screen. It was important to establish these results as a baseline to see the cases worked when there was maximised stimulus.

Second Equipment Test: Size

The purpose of the second test was to:

- Pilot stimulus overlay size to establish a range of sizes that provide adequate stimulation to elicit an SSVEP in the user.

This test was the first to use the DirectX and C++ build of the application. With this application it was possible to preconfigure the settings for the stimulus so that any participants could simply switch between the settings. An extended summary on this application may be found earlier in this document. The primary focus was to move away from having a full screen stimulus. Small, medium, and large stimuli were tested to provide a comparison. All tests were completed using 16Hz as a standard frequency, and two researchers too part to provide varied results. To summarise the tests completed were:

- Large white stimulus on video flashing at 16Hz
- Medium white stimulus on video flashing at 16Hz
- Small white stimulus on video flashing at 16Hz
- Large white stimulus on black flashing at 16Hz
- Medium white stimulus on black flashing at 16Hz
- Small white stimulus on black flashing at 16Hz

The following figures demonstrate the relative sizes to the on screen content used as stimulus. Figure 5 shows the large, Figure 6 shows the medium, and Figure 7 shows the small. The application is capable of showing at many different sizes, but these were selected for their variety and staggered difference in size.

Figure 5: Large flash on video

Figure 6: Medium flash on video

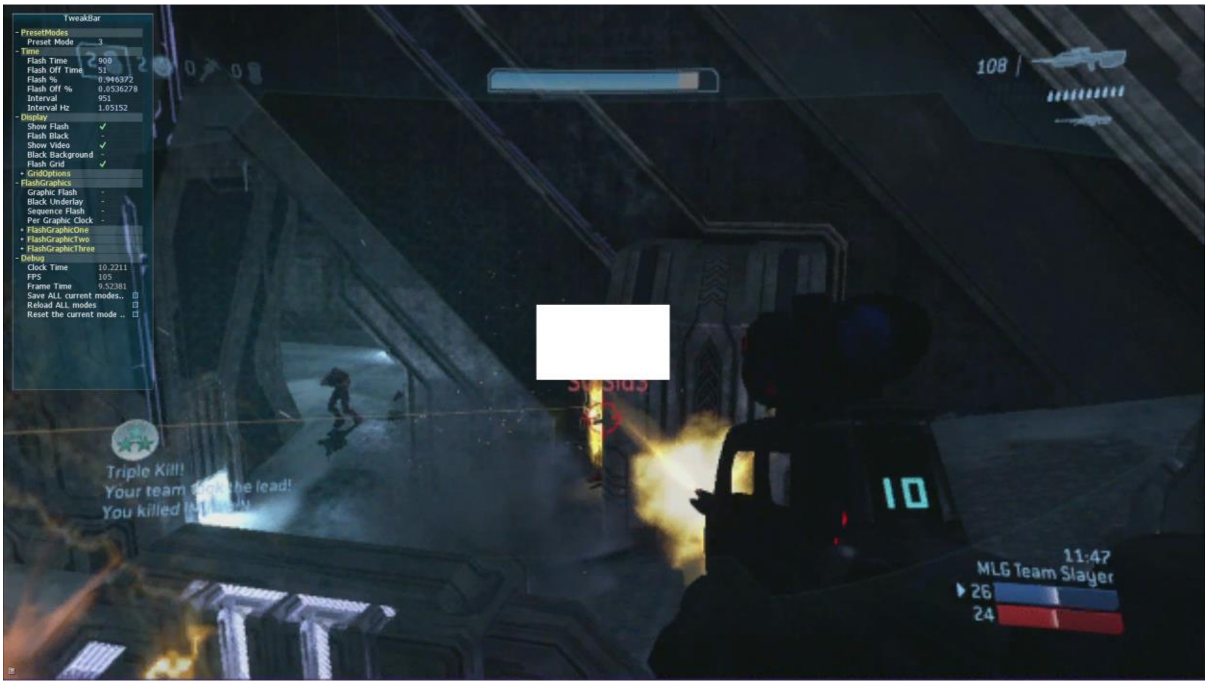

Figure 7: Small flash on video

Figure 8 shows the results for a single participant divided into classifications of the white on black and white on video. In the case of white on black the large and medium had the same spike at 15Hz, while the spike for small stimulus was smaller, but still a large SSVEP response. The responses were smaller for white on video probably due to the noise of the stimulus against the video and lower contrast than white on black. Strangely the medium gave a larger spike than the large for white on video. This was likely an error based on some unaccounted for change in equipment between tests. Experiments with a proper set of participants would be in a far more controlled environment and would be less prone to errors such as this occurring.

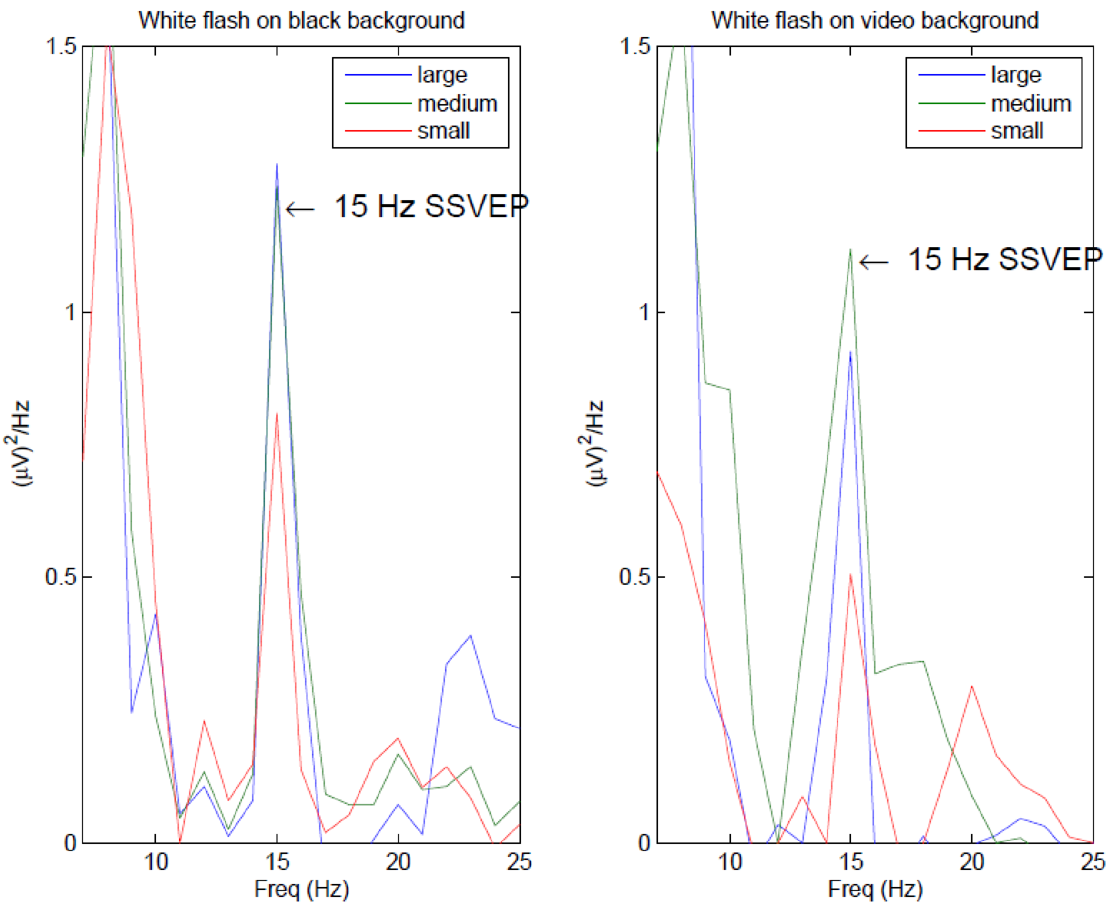

Figure 8: Second Equipment Test Results (Size)

Third Equipment Test: Stimulating Frequency

The purpose of the third test was to:

- Pilot stimulating frequency to establish a range of frequencies suitable for eliciting an SSVEP in the user.

For this test the frequencies used were 12Hz, 18Hz, and 22Hz. The stimulus used was the small white stimulus used during the second test. The purpose for using the small stimulus was to base any results on the expected size of stimulus that could be used in a game context. Games are typically made up of many smaller components, such as characters moving around on a screen. Therefore it was necessary to perform any further equipment tests using this size so that any future experiments could continue to utilise this. See Figure 7 for an example of the "small" stimulus and Figure 9 shows an example of white on black. Both the black and video backgrounds were used to provide a comparison. This meant a total of 6 cases were used and two researchers were shown the stimulus. To summarise, the following combinations were used:

- Small white stimulus on video flashing at 12Hz
- Small white stimulus on video flashing at 18Hz
- Small white stimulus on video flashing at 22Hz
- Small white stimulus on black flashing at 12Hz
- Small white stimulus on black flashing at 18Hz
- Small white stimulus on black flashing at 22Hz

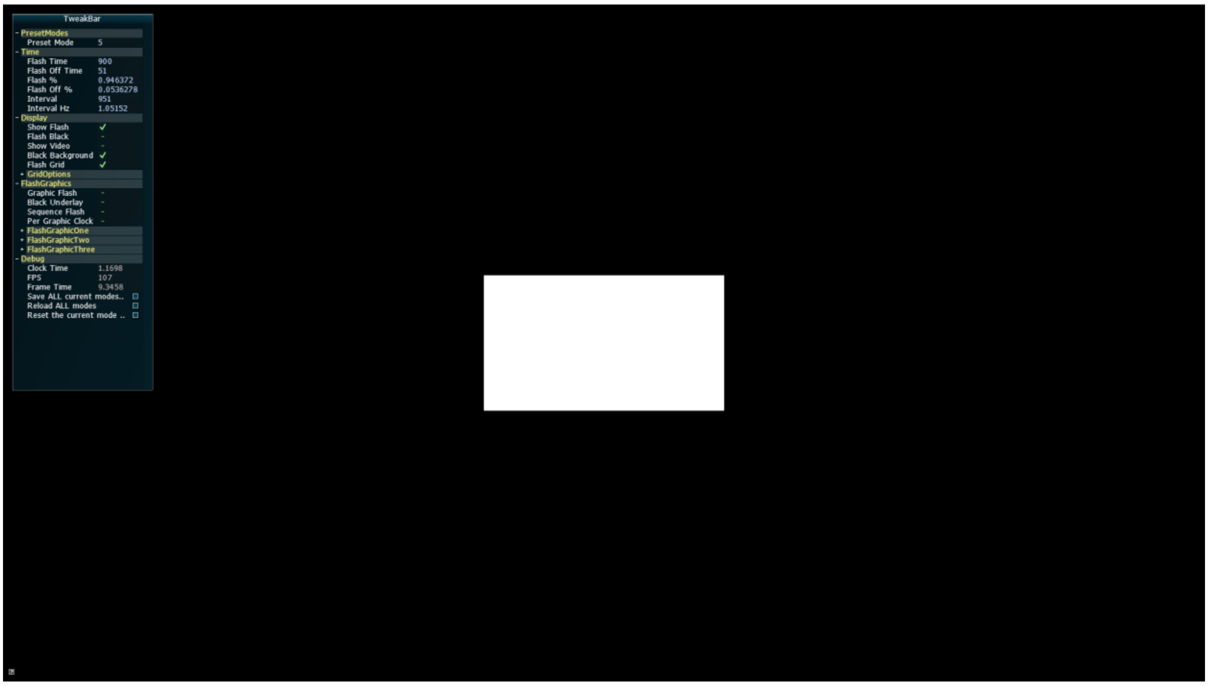

Figure 9: Medium flash on black

The results of the third test were inconclusive. An SSVEP was not elicited as significantly with the spike seen in the previous tests. A larger cohort of test participants would be required to properly establish a set of frequencies as being suitable for SSVEP in this context. The results of this test have been excluded for this reason.

Future Work

Now that the apparatus has been developed and tested, the next step is to conduct a full controlled experiment to fully evaluate the system and to establish its reliability and sensitivity across people. A limitation of the current stimulus apparatus is that with current computer systems commonly available it is not possible to flash the stimulus fast enough such that the user does not perceive the flashing (i.e. > 30Hz). It may be possible one day to have the stimulus flashed unperceivably, but for now it has been shown that the stimulus works, in limited testing, for basic test cases with a small flash on screen. Future work should include some of the following activities:

- A proper controlled experiment with participants to evaluate the system
- Development of the stimulus into example games that could provide some form of interactive interaction via the SSVEP.
- Testing to determine the difference between where a person is looking on the screen relative to stimulus. This would provide a basis for determining if the stimulus is merely visible or if it is an intended target of some user action.
- Testing to determine how possible it is to have multiple stimuli provided on the screen at the same time that could provide varying user actions.
- Testing to determine the responsiveness of using SSVEP and how that impacts on the speed of interaction.

The next phase for any experiment conducted should establish first that the colour, size, and frequency configurations used in this pilot study elicit SSVEP for a set of participants. Once these have been established as baselines for participants the above list of suggested tests could be conducted and compared to the baseline results.

Conclusion

This summer scholarship has developed a configurable stimulus apparatus necessary for ongoing SSVEP BCI research. As part of the development process, limited testing has been undertaken which suggest the apparatus performs to the required specification. The only limitation is that, due to computer hardware constraints, the flashing frequency is not able to be high enough such that the user does not perceive that flashing. Nonetheless, the pilot results indicate that SSVEP could potentially be used for interactive gameplay or other activities using flashes as stimuli provided by the computer monitor. Further work should seek to confirm this in a full controlled experiment. The design criteria for this summer scholarship outlined at the start of this document have been met. The stimulus elicited SSVEP while presented on a variety of different backgrounds including video. The application developed provided the required controls to change the rate of flashing and size in addition to many other parameters that could be used for constructing future tests.