An Investigation of Triggering Approaches for the Rapid Serial Visual Presentation Paradigm in Brain Computer Interfacing

Zhengwei Wang[1], Graham Healy[2], Alan F. Smeaton[2], and Tomas E. Ward[1]
[1]Department of Electronic Engineering, Maynooth University, Maynooth, Ireland
[2]School of Computing, Dublin City University, Dublin, Ireland
Email: wang.zhengwei.2014@mumail.ie

***Abstract*—The rapid serial visual presentation (RSVP) paradigm is a method that can be used to extend the P300 based brain computer interface (BCI) approach to enable high throughput target image recognition applications. The method requires high temporal resolution and hence, generating reliable and accurate stimulus triggers is critical for high performance execution. The traditional RSVP paradigm is normally deployed on two computers where software triggers generated at runtime by the image presentation software on a presentation computer are acquired along with the raw electroencephalography (EEG) signals by a dedicated data acquisition system connected to a second computer. It is often assumed that the stimulus presentation timing as acquired via events arising in the stimulus presentation code is an accurate reflection of the physical stimulus presentation. This is not necessarily the case due to various and variable latencies that may arise in the overall system. This paper describes a study to investigate in a representative RSVP implementation whether or not software-derived stimulus timing can be considered an accurate reflection of the physical stimuli timing. To investigate this, we designed a simple circuit consisting of a light diode resistor comparator circuit (LDRCC) for recording the physical presentation of stimuli and which in turn generates what we refer to as hardware triggered events. These hardware-triggered events constitute a measure of ground truth and are captured along with the corresponding stimulus presentation command timing events for comparison. Our experimental results show that using software-derived timing only may introduce uncertainty as to the true presentation times of the stimuli and this uncertainty itself is highly variable at least in the representative implementation described here. For BCI protocols such as those utilizing RSVP, the uncertainly introduced will cause impairment of performance and we recommend the use of additional circuitry to capture the physical presentation of stimuli and that these hardware-derived triggers should instead constitute the event markers to be used for subsequent analysis of the EEG.**

***Index Terms*—BCI, LSL, RSVP, Trigger.**

I. INTRODUCTION

Brain computer interfaces (BCIs) provide a non-muscular approach for the user to communicate with others or to control external devices. One of the original goals of such systems has been to provide users, who may be completely paralyzed, or ‘locked in’, with basic communication capabilities so that they can express their wishes to caregivers or even operate word processing programs or neuroprostheses[1]. General BCI platforms are suitable for such investigations with a variety of brain signals derivations, processing methods and applications currently available[2]. More recently BCI for augmenting or enhancing human computer interaction more generally have emerged. These systems can monitor brain state while participants are engaged in everyday tasks (but most commonly involves interaction with a computing device) and the output is used to improve the user’s experience or their performance in some way. The work in this paper is contextualized by one such implementation and associated paradigm. We are developing BCI applications capable of retrieving images from large image datasets, enhancing the performance of visual search through a high throughput image presentation rate. An advanced EEG-based neurotechnology has been demonstrated, to monitor participants’ brain responses while they are engaged in visual search through a rapidly presented series of images [3]. A suitable paradigm to improve the utility of such an approach for a user is through a RSVP BCI paradigm.

The topic of rapid serial visual presentation (RSVP) is introduced using a familiar example, that of rapidly riffling through the pages of a book in order to locate a needed image[4]. RSVP has been divided into two main categories where one is text RSVP and the other one is image RSVP. In image RSVP, rapid succession of images are presented to a participant on a monitor. From experiments such as those by Potter[5], it has been shown that the target begins to be missed by participants with presentation times under 125 ms. Therefore, the presenting rate of RSVP is suggested not to exceed 8 Hz. RSVP related EEG aims to develop a kind of BCI application which is related to human beings’ perception, visual system and so on. Most of current BCIs use event-related potentials (ERPs) in EEG as inputs, which are neural signatures representing the responses to an external stimulus. ERPs are EEG changes that are time locked to sensory, motor or cognitive events that provide a safe and noninvasive approach to study psychophysiological correlates of mental processes[6]. The P300 component, a type of ERP, is elicited with visual stimuli in subjects and is typically presented between 300 ms - 600 ms after the appearance of a rare visual target within a sequence of frequent irrelevant stimuli[7]. Thus targets in the presented image stream are not suggested to be placed close to each other in order to generate a reliable P300 component in EEG. In real world search applications, however, the distribution and ordering of target stimuli may be unknown. The framework for RSVP based BCI usually uses amplitude measures of P300 component activity extracted from EEG epochs across multiple channels as features to build suitable classifier models[8]. This indicates that timing properties of recorded RSVP related EEG epochs are essential to the classifier model for BCI.

There are different EEG acquisition systems in the market and they are often incompatible with each other, which prevents unified collection of recorded EEG data and unified BCI applications. While systems like BCI2000 exist to deal with this issue of underlying system heterogeneity such a system might be too heavy weight in some instances for modern BCI application requirements. An alternative approach is a stack such as Lab Streaming Layer (LSL). LSL is a system for the unified collection of measurement time series data in research experiments that handles both the networking, time-synchronization, real-time access as well as optionally the centralized collection, viewing and disk recording of the data. LSL enables the physical separation of the stimulus presentation machine and the recording machine in RSVP experiments. In this study, we will use LSL to capture measurements that will allow us to evaluate the latency between the image presentation in software and the image time of the physical presentation on screen. Furthermore, we can determine the LSL time synchronization performance across two different computers. The software trigger is implemented in Python by using the simulation and neuroscience application platform (SNAP). The hardware trigger is generated by means of a light diode resistor comparator circuit (LDRCC).

II. TRADITIONAL RSVP RELATED EEG DATA ACQUISITION PARADIGM

RSVP related EEG data acquisition is normally deployed on two computers. One is for stimuli presentation to participants and the other is for recording EEG data from participants. As shown in Fig. 1 in our example, a BioSemi EEG amplifier is used for EEG signals recording and amplification while a BioSemi stimulus box is used for converting the digital optical data coming from the amplifier to an USB2 output. The traditional RSVP paradigm involves connecting a presentation machine to the EEG acquisition system via a suitable channel such as a parallel cable. The presentation machine uses a software program (e.g. PsychoPy) to display image streams to the participant and generates software triggers which mark the onset of target and distractor images. Target images are visually separable inside the stream from the rest of the distractor images using some given criteria. These triggers are sent to EEG acquisition system via a suitable connection (i.e. a parallel cable in this example). This is the traditional approach of sending software triggers to EEG acquisition system. Then the RSVP-related EEG data which contains triggers and raw EEG data are collected by the data acquisition computer. This traditional data acquisition paradigm is limited by the type of EEG acquisition system used and requires a suitable data transmission connection between presentation machine and EEG acquisition system which is not flexible.

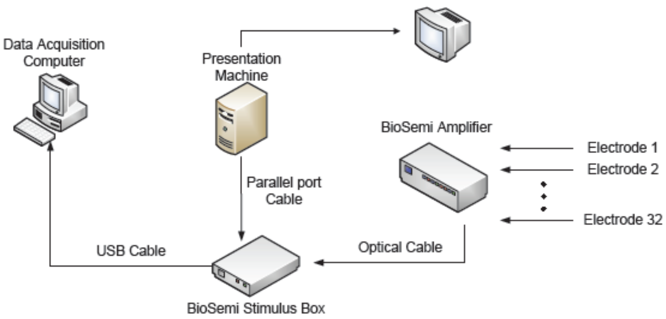

Fig. 1. Traditional RSVP related EEG data acquisition paradigm.

III. RSVP RELATED EEG DATA ACQUISITION VIA LSL

Abstracting the underlying hardware (e.g. amplifiers, events triggers, stimuli presentation equipment) can lead to more flexible and more readily deployable and shareable BCI systems. A modern set of software components which is popular for this purpose is the LSL. LSL records data as a streaming type, which is compatible with many modern EEG acquisition systems and allows multiple computers to sync data stream recording. As shown in Fig. 2, the EEG acquisition system records hardware triggers and EEG data where these two signals are then acquired by data acquisition machine. This kind of data is then sent to LSL. Software triggers are sent via LSL by the RSVP presentation machine directly. EEG data and trigger signals in LSL can be saved as a unified .XDF file. This type of file is imported to MATLAB or other suitable software for subsequent processing of the data.

A. *Experimental Setup*

The experiment described in this paper included two main parts: 1. Exploring the latency between image presentation in software and physical image presentation in LSL on a single computer; 2. Exploring the performance of LSL time synchronization for two different computers. Each experimental trial consisted of 50, 100, 200, 500, and 1000 image events. The BioSemi Active View 2 system AD box sampled data at a customized 512 Hz sampling rate via optical connection to BioSemi amplifier and via USB connection to a data acquisition computer. The API in LSL was driven through a software application called “Lab Recorder” which was used to record data streaming (EEG data, hardware and software triggers) in LSL and the recorded data streaming was saved as a .XDF file. The .XDF file was then imported to MATLAB as part of data analysis.

Experimental results in this paper strongly depend on the respective software that is used for presenting images and the type of processor, graphic card, etc. in the presentation machine. Results in this experiment can be used as reference

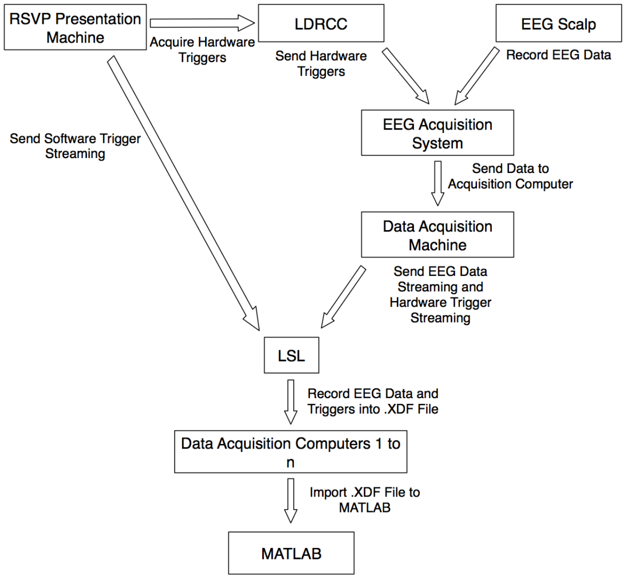

Fig. 2. RSVP related EEG data acquisition via LSL diagram.

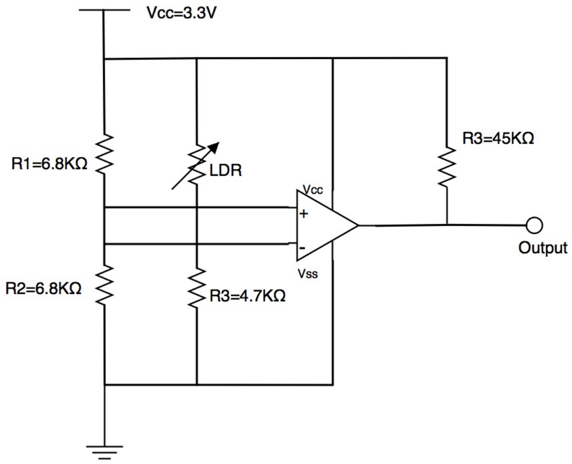

Fig. 3. LDRCC architecture.

for seeing the types of timing differences that arise between software triggers and hardware triggers. The presentation machine used in this experiment was a Dell XPS 8700 desktop which uses a 4th generation Intel core i7-4790 processor, AMD Radeon HD R9 270 2GB GDDR5 video card and Dell 2313H monitor.

B. LDRCC Architecture

The LDRCC was built in order to explore the latency between image presentation in software and physical image presentation in LSL. Because the response time of a circuit generating the digital output is relatively instantaneous, the hardware trigger is capable of capturing physical image presentation time. Fig. 3 shows the architecture of the LDRCC. A light diode resistor (LDR) used in this circuit revealed high impedance up to 9 KΩ corresponding to low level light and low impedance down to 1.5 KΩ corresponding to high level light. R_3 was a pull-up resistor that was used to drive the trigger IO in EEG acquisition system. The LDRCC used LM311 single comparator, whose response time takes approximately 200 ns, to generate the digital output (high voltage was 3.3 v and low voltage was 0 v).

C. Trigger Acquisition

1) Hardware trigger acquisition: In order to generate hardware triggers in RSVP experiment, the LDR in LDRCC was taped on the top-left corner of the presentation monitor. Black and white images started to appear alternately in that region at the onset of image stimuli with the changing rate at the same rate as presented images shown on the monitor. When the presented image changes each time, the LDRCC generates a rising or falling edge, which marks the onset and the end of the presented image. There are two reasons choosing only black and white images in that region: 1. It aims to give the same light change to LDR regardless of presented images; 2. Light change between while and black images is larger than any other two colors. Hence, this light change enables the largest change in the resistance of LDR, which in turn makes the largest voltage change at the positive input of comparator. It is faster for comparator to generate triggers for the changing of presented images. Hardware triggers were then sent to BioSemi Active View 2 system. LDRCC output was connected to the pin 16 of trigger IO at the back of BioSemi stimulus box and other pins were connected to the ground. Capturing both EEG and triggers in this way on a single data acquisition device allows for the highest precision in time alignment.

2) Software trigger acquisition: Image presentation and software trigger generation were implemented in Python using SNAP. Software triggers were generated directly prior to the execution of image presentation code. Triggers were sent to LSL via SNAP.

Fig. 4 shows 20 hardware and software triggers generated from 20 presented images. The rate of image streaming is 5 Hz (time interval between each image is 0.2 s). The blue line is the hardware trigger signal while red line is the software trigger signal. It can be seen that hardware trigger signal is a square wave where the high voltage is 3.3 V and low voltage is 0 V. The amplitude of the software trigger does not have any meaning in terms of experimental interpretation. The default value of LDRCC output is 0 V at the beginning and it can be seen that the software trigger precedes the hardware trigger in this case.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

There are three factors resulting in the latency between collected hardware and software triggers in LSL: 1. Latencies between image presentation in software and physical image presentation; 2. Time synchronization performance if using different computers for recording data and image presentation. 3. Network latency arising because software and hardware triggers are sent to network independently for both 1-computer

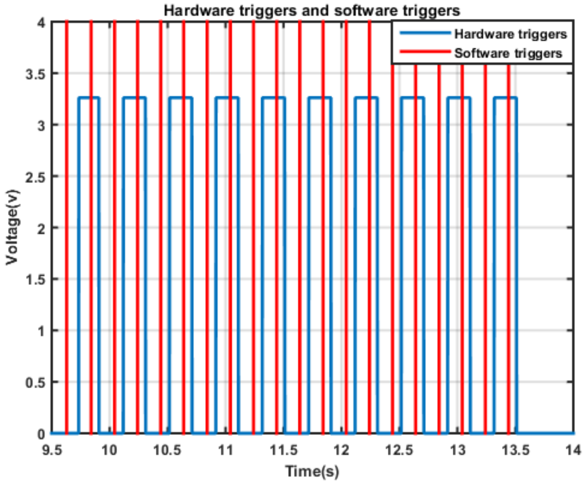

Fig. 4. Hardware and software triggers captured using a 2-machine setup (presentation & data acquisition).

and 2-computer implementations. A CPU needs some time to execute image presentation in software and display an image on the monitor, which causes the first latency. Because LSL records the timestamps of recorded data corresponding to the local computer, the second factor is caused by the LSL time synchronization of different computer clocks in network layer. The third factor can not be avoided even using a high speed network so we only investigated the first two factors in this paper. Firstly, we investigated the first factor via comparing the time differences of corresponding hardware and software triggers. Because software triggers and hardware triggers are sent to LSL independently, the network latency can have different effects on these two types of triggers, however, we assume this latency to be constant and to be effectively cancelled out when we subtract corresponding hardware/software timestamps when comparing delays between these trigger types.

Therefore, the latency here represents the difference between image presentation in software and physical image presentation (network latency can be almost neglected). Before validating the LSL time synchronization performance, it is necessary to minimize the first latency factor. Hence, we used data, which contains minimum latency error between image presentation in software and physical image presentation (i.e. Group 50/100), to explore the time synchronization performance of LSL.

A. Latencies between image presentation in software and physical image presentation

The latency between image presentation in software and physical image presentation was attained by t_h - t_s, where t_h and t_s are hardware and software timestamps. Fig. 5 shows histograms of these latency values for 50, 100, 200, 500 and 1000 images group cases. The reason we did this was to assess differences in recorded presentation times in software and physical image presentation times.

We calculated statistical characteristics of each distribution (see Table I). As the distributions seen in Fig. 5 appear non-gaussian (particularly for Group 50 & 100), we use a median statistic for reporting in Table I. Examining Table I and Fig. 5 in tandem we can see that increasing image numbers negatively impacts (i.e. increases) our median latencies. These statistical characteristics show that the first two groups have smaller latency errors between image presentation in software and physical image presentation compared to the last three groups. The increasing latency encountered for increasing image count is in all likelihood caused by the software implementation. When implementing RSVP experiments in some softwares for large datasets, it is necessary to make efforts such as prebuffering images into memory and/or unallocating memory for each loaded image after it is presented. Without employing such efforts, the presentation software may exhibit issues such as slowing presentation speed as each image is loaded but not removed from memory after presentation. Such overheads in turn can cause other operating system functionality and/or network functionality to be impeded, giving rise to a range of other complex effects that can in turn potentially affect timing characteristics of the presentation software further. We calculated the median latency values of the first 50 points and the last 50 points for 100, 200, 500 and 1000 image groups in order to see whether potential software problems were causing larger latencies with increasing time. From the last three columns in Table I, it can be seen that such software implementation problems are the cause of larger latencies in the group of 200 images, 500 images and 1000 images but the 100-image group does not suffer from these types of software problems. Therefore, we conclude that the median difference of 0.025 s is a realistic approximation of the real difference between image presentation in software and physical image presentation in this RSVP implementation for our system.

TABLE I
TIME-RELATED LATENCIES BETWEEN IMAGE PRESENTATION IN SOFTWARE AND PHYSICAL IMAGE PRESENTATION OF DIFFERENT GROUPS USING 1-MACHINE SETUP

Group	50 images	100 images	200 images	500 images	1000 images
MV (s)	0.0255	0.0250	0.0295	0.0369	0.0638
First 50 points (s)	0.0255	0.0270	0.0250	0.0360	0.0520
Last 50 points (s)	0.0255	0.0240	0.0300	0.0410	0.0630

Notably, however, in Fig. 5 we can see that Group 50, Group 100 and Group 200 are bimodal distributions. In order to evaluate what might be causing this we: 1) used a median split to firstly divide the latencies into lower and upper ranges, 2) Applied further median splits to these two new ranges for Group 50 and Group 100, 3) Calculated the difference between these respective upper and lower median splits. This can be seen in Fig. 5 for Group 50 and Group 100 where the lower

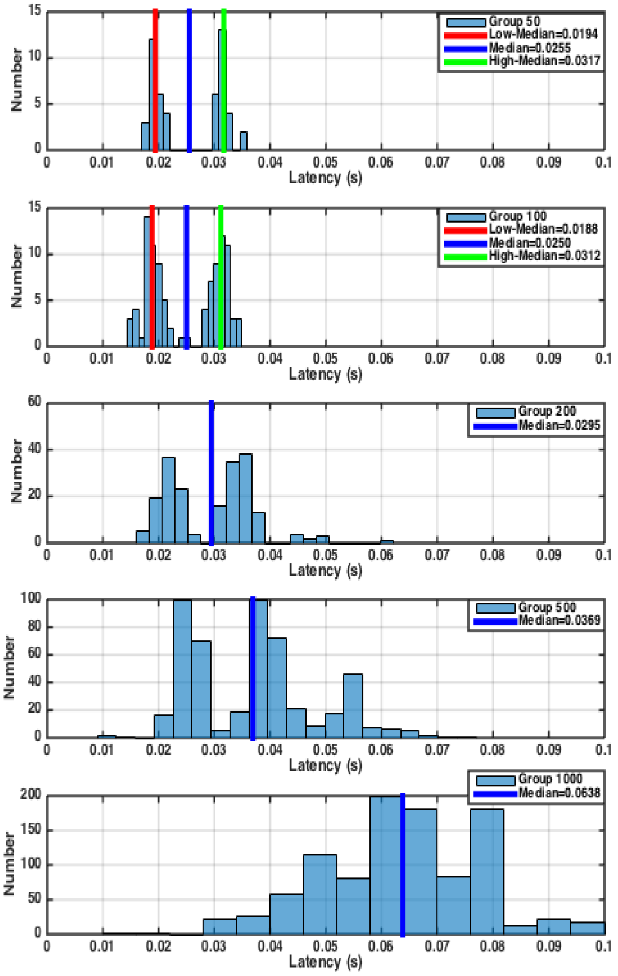

Fig. 5. Histograms of latencies derived from (paired) differences between hardware and software trigger timestamps (hardware timestamp - software timestamp), as a way to assess timing differences arising due to issues such as those caused in software implementation. We show distributions for groups of 50, 100, 200, 500 and 1000 images using a 1-machine setup (respectively row by row). Shown in blue vertical lines are median values. In the first two histograms (i.e. Group 50/Group 100) in red and green we can see vertical lines corresponding to median values for lower and upper ranges after a median split.

median split is in red and the upper median split is in green. What we find is that there is a 0.0121 second and 0.0124 second difference between these upper and lower medians for Group 50 and Group 100 respectively. In effect, we can say there is an additional latency affecting half of our trigger samples that is between 0.0121 second and 0.0124 second.

In Fig. 6 for the Group 100 case (where we examine time intervals for both hardware and software triggers), we can see that there is relative stability to the frequency of software triggers where hardware-sensed triggers are seen to be more variable (and bimodal). As we are using the difference of these relative timestamps to generate Fig. 5, we identify this as causing the bimodal distributions we see in Fig. 5. These differences are likely related to the refresh rate of the monitor used where there is an approximate 50% likelihood that the stimulus presentation will not happen until the next refresh. These variable timing differences (0.0121 s and 0.0124 s for Group 50/100 respectively) are relatively close to the refresh time interval of the monitor (0.0167 s i.e. 60 Hz) used in our experiment.

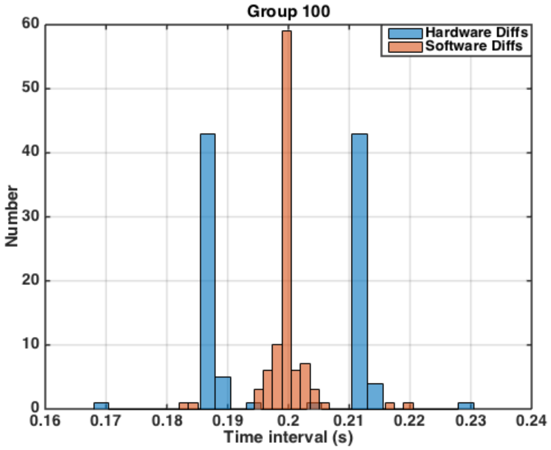

Fig. 6. Distribution of interval differences in timestamps for hardware triggers (in blue) and software triggers (in orange).

B. *Evaluating the LSL time synchronization performance*

LSL time synchronization performance was evaluated using three steps:

1. Calculating different values using hardware trigger timestamps - software trigger timestamps on a single machine: $(t_h - t_s)_{1-computer}$;

2. Repeating the process but this time using different stimulus presentation and data acquisition computers: $(t_h - t_s)_{2-computer}$;

3. Calculating the mean difference of values which are calculated in previous two steps and this mean difference can be regarded as LSL time synchronization performance: $\text{mean}((t_h - t_s)_{2-computer}) - \text{mean}((t_h - t_s)_{1-computer})$.

Because the latency between software triggers and hardware triggers for 2-computer setup (latency can be seen in Fig. 4) is affected by the latency error between image presentation in software and physical image presentation and the LSL time synchronization performance (assuming network latency has the same effect on 1-computer and 2-computer implementations), it is necessary to get rid of the first latency error before evaluating the LSL time synchronization performance. From previous section, it shows that latencies between image presentation in software and physical image presentation of Group 50 and Group 100 are smaller comparing to others. Hence, we used these two groups in this part of the evaluation.

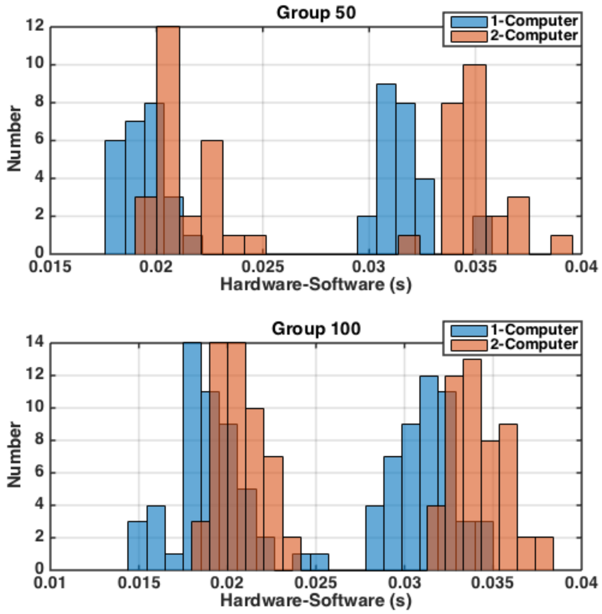

Fig. 7. Distribution of hardware trigger timestamps - software trigger timestamps for 1-computer and 2-computer.

Fig. 7 shows the distribution of hardware trigger timestamps - software trigger timestamps (t_h - t_s) for 1-computer and 2-computer cases. This indicates that t_h - t_s consists of the difference between image presentation in software and physical image presentation. It can be seen that the histogram has two sets of data distributions (bimodal) for each comparison. As the network latency can be regraded as a constant approximately that is cancelled out when taking the difference of these trigger types, differences between hardware triggers and software triggers for 1-computer case only depend on the distribution of differences between image presentation in software and physical image presentation. We calculated the mean difference of 1-computer and 2-computer for each comparison in Fig. 7 to evaluate the LSL time synchronization performance between two computers. Mean differences for Group 50 and Group 100 are 0.0026 s and 0.0025 s respectively and these two MVs are much smaller than those in evaluating difference between image presentation in the software and physical image presentation part. This indicates that LSL time synchronization performance is good and it can be used for implementing RSVP among different computers.

V. CONCLUSION

This paper investigated timing discrepancies in stimulus presentation timing when relying on software only timing information. Hardware in the form of light detection circuits were used to provide accurate timing information on stimulus presentation and this was compared to events generated in the corresponding software (captured in this case via LSL) for an RSVP experiment. Results demonstrate that the latency exists between the image presentation in software and physical image presentation even for 50 and 100 images group and software problems arise with increasing image datasets (i.e. starting from 200 images). It should be stressed that this is due to software problems (e.g. crippling memory overhead) of presenting images and the refresh time interval of the monitor instead of LSL which performs well in time synchronization. LSL is simply used for syncrhonized data capture. We suggest that for RSVP protocols where temporal accuracy is important that unless demonstrated otherwise a hardware solution for monitoring physical presentation of images should be used.

ACKNOWLEDGMENT

This work was a part of the Insight Centre for Data Analytics which is supported by Science Foundation Ireland under Grant Number SFI/12/RC/2289.

REFERENCES

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, “Brain–computer interfaces for communication and control,” *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.

[2] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, “Bci2000: a general-purpose brain-computer interface (bci) system,” *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1034–1043, 2004.

[3] N. Bigdely-Shamlo, A. Vankov, R. R. Ramirez, and S. Makeig, “Brain activity-based image classification from rapid serial visual presentation,” *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 16, no. 5, pp. 432–441, 2008.

[4] R. Spence and M. Witkowski, *Rapid serial visual presentation: design for cognition*. Springer, 2013.

[5] M. C. Potter, “Short-term conceptual memory for pictures.,” *Journal of experimental psychology: human learning and memory*, vol. 2, no. 5, p. 509, 1976.

[6] S. Sur and V. Sinha, “Event-related potential: An overview,” *Industrial psychiatry journal*, vol. 18, no. 1, p. 70, 2009.

[7] R. G. O’Connell, J. H. Balsters, S. M. Kilcullen, W. Campbell, A. W. Bokde, R. Lai, N. Upton, and I. H. Robertson, “A simultaneous erp/fmri investigation of the p300 aging effect,” *Neurobiology of aging*, vol. 33, no. 10, pp. 2448–2461, 2012.

[8] Y. Huang, D. Erdogmus, M. Pavel, S. Mathan, and K. E. Hild, “A framework for rapid visual image search using single-trial brain evoked responses,” *Neurocomputing*, vol. 74, no. 12, pp. 2041–2051, 2011.