Synchronization of acquisition devices in neuroimaging: An application using co-registration of eye movements and electroencephalography

Gelu Ionescu, Aline Frey, Nathalie Guyader, Emmanuelle Kristensen, Anton Andreev, Anne Guérin-Dugué

Synchronization of acquisition devices in neuroimaging:
An application using co-registration of eye movements and electroencephalography

Gelu Ionescu[†], Aline Frey[2], Nathalie Guyader[1], Emmanuelle Kristensen[1], Anton Andreev[1], & Anne Guerin-Dugué[1]

[1]Univ. Grenoble Alpes, CNRS, Grenoble INP*, GIPSA-lab, 38000 Grenoble, France

* Institute of Engineering Univ. Grenoble Alpes

[2]Laboratoire de Neurosciences Cognitives, UMR 7291, CNRS - INSPE d'Aix-Marseille Université, France

Corresponding author:

Anne Guérin-Dugué, GIPSA-lab,

11 rue des Mathématiques,

Grenoble Campus BP46,

F-38402 Saint Martin d'Hères Cedex

E-mail : anne.guerin@gipsa-lab.grenoble-inp.fr

Tel: +33 (0)4 76 57 47 90

Abstract

Interest in applications for the simultaneous acquisition of data from different devices is growing. In neuroscience for example, co-registration complements and overcomes some of the shortcomings of individual methods. However, precise synchronization of the different data streams involved is required before joint data analysis. Our article presents and evaluates a synchronization method which maximizes the alignment of information across time.

Synchronization through common triggers is widely used in all existing methods, because it is very simple and effective. However, this solution has been found to fail in certain practical situations, namely for the spurious detection of triggers and/or when the timestamps of triggers sampled by each acquisition device are not jointly distributed linearly for the entire duration of an experiment. We propose two additional mechanisms, the "Longest Common Subsequence" algorithm and a piecewise linear regression, in order to overcome the limitations of the classical method of synchronizing common triggers.

The proposed synchronization method was evaluated using both real and artificial data. Co-registrations of electroencephalographic signals (EEG) and eye movements were used for real data. We compared the effectiveness of our method to another open source method implemented using EYE-EEG toolbox. Overall, we show that our method, implemented in C++ as a DOS application, is very fast, robust and fully automatic.

Keywords

Co-registration, Synchronization, Clock drift, Drift correction, Electroencephalography, Eye movements.

Synchronization of acquisition devices in neuroimaging:

An application using co-registration of eye movements and electroencephalography

Introduction

In neuroimaging, simultaneous recordings are a powerful way of investigating brain activity. All experimental modalities have their own advantages and limitations. They also have a certain degree of complementarity, mainly in the temporal domain (e.g. for EEG + eye movements, and in both the temporal and spatial domains (e.g., for EEG + fMRI), which has motivated intense efforts towards their combination (Jorge et al., 2015; Rosenkranz & Lemieux, 2010). Two main approaches coexist. The first approach - data integration - uses one modality to improve another, for example, when the high spatial resolution of functional Magnetic Resonance Imaging (fMRI) is used to constrain electroencephalography (EEG) inverse solutions for source localization (Liu, Ding & He, 2006). The second approach - data fusion - requires synchronous acquisition, and refers to the idea of using both modalities in a combined joint analysis, for example for the simultaneous recording of EEG and near-infrared spectroscopy (Shin, von Lühmann, Kim, Mehnert, Hwang & Müller, 2018), or for EEG and eye movements (Nikolaev, Meghanathan & van Leeuwen, 2016).

Ideally, one single data acquisition system would record all necessary data types, inherently time-synchronizing all measurements. In many cases however, multiple data acquisition systems are involved, and efficient data synchronization is needed to harmonize data continuously over time. Off-line synchronization methods rely on matching information from two or more simultaneous measurements (Hoogeboom, 2003). As a result of the synchronization, all data are sampled in the same time referential and share the same events.

In this context of data fusion, we propose a solution for the synchronization of data from different acquisition devices, and we compare it to a currently existing solution in terms of limitations and advantages. We illustrated our method by testing it on real research data obtained during the co-registration of EEG signals and eye movements. In this type of experimental setup, the EEG signal is segmented based on eye movement events, by using the temporal and spatial positions of specific eye events (e.g., fixations or saccades) as markers for brain signals. These EEG analyses, triggerred by eye events, are called Eye Fixation or Saccade Related Potentials (EFRP / ESRP), and can be analyzed using efficient methods of estimation (Kristensen, Guerin-Dugué & Rivet, 2017; Kristensen, Rivet & Guérin-Dugué, 2017; Ehinger & Dimigen, 2019). This co-registration technique is an effective way of delving into cognitive processes (Frey, Lemaire, Vercueil & Guérin-Dugué, 2018). It also allows for more ecological experimental protocols, which are not constrained by "external" markers, such as the onset of a word or image, and in which participants can perform fixations at their own pace (Dimigen, Sommer, Hohlfeld, Jacobs, & Kliegl, 2011; Körner, Braunstein, Stangl, Schlögl, Neuper & Ischebeck, 2014; Van Humbeeck, Meghanathan, Wagemans, van Leeuwen & Nikolaev, 2018).
It is being used increasingly to study attention, memory encoding, visual scene processing, reading, and responses to emotional visual information (for reviews, cf. for example (Dimigen, et al., 2011; Nikolaev, Pannasch, Ito & Belopolsky, 2014; Nikolaev et al., 2016). This type of joint acquisition therefore has a strong added value for the understanding of the time course of neural activity during cognitive tasks requiring a high degree of accuracy in the synchronization of timings.

Where systems of acquisition are concerned, suppliers of both eye-trackers and EEG systems have recently proposed additional modules which permit the importation and synchronization of co-registered data streams. The EEG analysis software ‘BrainVision Analyzer’ (Brain Products GmbH) proposed a first

version of the "Add Channels" module, supporting four types of eye-tracker data: Tobii (TobiiStudio223 and TobiiStudio301), ASL (ASLEyeTrac6 and ASLResults243), SMI (SMIBeGaze300) and SR Eye Link 1000 Plus. The whole process is based on common triggers, found by the module, for the drift correction. Gaze positions are then resampled and added to EEG channels (Brain Products Press Release, 2013[1]). The Tobii system proposed the Pro Lab software which uses Transistor-Transistor Logic (TTL) technology to send events from the parallel port to the other devices for synchronization[2]. Finally, iMotions combined Eye Tracking glasses with a wireless EEG device. In addition to these corporate solutions, which need to be purchased separately, the open source EEGLab environment (Delorme & Makeig, 2004) has developed a plug-in for the joint analysis of EEG signals and eye-tracker data (EYE-EEG[3], Dimigen et al., 2011). This plug-in also resolves the synchronization issue between these devices thanks to common triggers in the two data flows.

Thus, although several solutions exist, no consensus on an effective synchronization process seems to have emerged, and in general, each study attempts to find its own solution. However, it appears that in most cases, common triggers are sent as TTL pulses, on a parallel port, typically at the beginning of each trial (Dimigen et al., 2011; Guérin-Dugué, Roy, Kristensen, Rivet, Vercueil, & Tcherkassof, 2018; Van Humbeeck et al., 2018). Moreover, due to several potential problems, which will be discussed in this article, this synchronization procedure requires realignment precautions, and it is important to ensure its effectiveness. As mentioned above, the point of these co-registration methods is often to obtain good

[1] https://www.brainproducts.com/analyzer203.php?tab=4
[2] https://www.tobiipro.com/learn-and-support/learn/steps-in-an-eye-tracking-study/design/solutions-for-co-registration-of-eye-tracking-and-other-biometric-measures/
[3] http://www2.hu-berlin.de/eyetracking-eeg/

temporal resolution and a few milliseconds of lag can be very damaging to the analysis and understanding of the results.

As an example, Kamienkowski, Ison, Quiroga, and Sigman (2012)evaluated the synchronization procedure by comparing the onset of saccades detected by electro-oculography (EOG) channels to the onset of saccades detected by the Engbert and Kliegl (2003) algorithm in the eye-tracking recording, which showed lags between saccade onsets of under 20 ms in all trials. In the latest version of the EYE-EEG[3] toolbox, synchronization accuracy can be checked by cross-correlation of eye positions and EOG channels for both horizontal and vertical movements. For successful synchronization, the lag that maximizes the cross-correlation must be close to zero.

In the following sections, we describe the framework for synchronizing independent acquisition devices and the practical difficulties that can be encountered. We implemented the proposed method in a software named “GazeEEGSynchro”, written in C++ and implemented as a DOS application, that is provided and available in the Zenodo repository (https://doi.org/10.5281/zenodo.5554071) with examples on real data. We thus describe our proposed approach, validated with simulated and real data -acquired by our research team and by a team from another laboratory-, in order to evaluate its effectiveness, and we compare it with the EYE-EEG toolbox.

Trigger-based synchronization

When processing recorded signals off-line, it is essential to readjust these signals into the same time referential because of the inaccuracy of the crystal-based clock of each individual device (clock drift issue). Trigger-based synchronization is the simplest and most commonly used solution for the synchronization of devices working at the same or at different sampling rates. In this section, we begin by describing the clock

drift issue between the oscillators of each acquisition device, and the synchronization principle usually employed to resolve this.

Clock drift issue

In an acquisition system with several acquisition devices, each device's clock is controlled by its own crystal oscillator. Even when two quartz crystals come from the same batch, oscillators can exhibit a variety of instabilities related to aging, frequency, changes in temperature, power supply inaccuracy, and wire interconnection lengths. The phenomenon whereby a clock does not run at the same nominal speed as an ideal clock is called clock drift (Pak, 2017). In practice, quartz crystals are manufactured for frequencies ranging from a few tens of kHz to tens of MHz, and are often designed around standard frequencies[4] such as 3.579545 MHz adopted by the National Television System Committee, 10 MHz for low-power microcontrollers, 33.33 MHz or 40 MHz for computers. A programmable frequency divider is also necessary, because in acquisition systems the desired sampling frequency should be programmable in a range of tens of Hz to several tens of kHz. Therefore, the clock drift of the operating frequency also has an impact on these frequency dividers. Consequently, efficient synchronization is needed to compensate for the relative differences in timing generated by acquisition systems affected by different degrees of clock drift.

[4] https://en.wikipedia.org/wiki/Crystal_oscillator_frequencies

Principle of trigger-based synchronization

We describe the trigger-based synchronization aimed at solving the clock drift issue using a very simple and generic experimental design which involves two acquisition devices and a master computer, as depicted in *Figure 1*.

The STIMULATION PC (master) is used to present external stimuli (visual, auditory or both) to the participant, to record if necessary their response, and send triggers through a logical bus (TTLB). The triggers are used to tag the different stages of the experiment (start, end, visual display, sound emission, etc.), and they are received instantly by all acquisition devices. Two acquisition devices (ACQD1 and ACQD2), controlled by two independent PCs (HOST PC1 and HOST PC2), complete this set-up. The STIMULATION PC communicates with the two HOST PCs through two control buses (CB1 respectively CB2; i.e. Ethernet, USB, etc.). Likewise, the two HOST PCs communicate with the acquisition devices through two control/data buses (CDB1 and CDB2, i.e. Ethernet, USB or PCI). These buses are used to control the acquisition devices and to transfer the acquired data (digitized analog signals and TTL triggers). In this way, data are continuously stored in two different files on the corresponding HOST PCs.

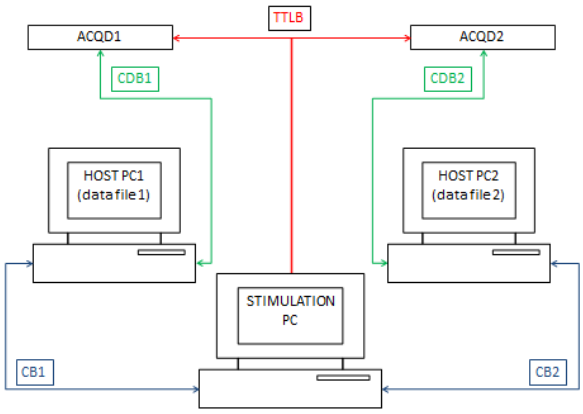

Figure 1. Example of an acquisition system with two acquisition devices: ACQD1 and ACQD2; three PCs: STIMULATION PC (master), HOST PC1 and HOST PC2 (acquisition devices); two bidirectional

control/data buses CDB1 and CDB2 (communication between each host PC and the acquisition device); two bidirectional control buses CB1 and CB2 (communication between the STIMULATION PC and the two HOST PCs); a logical bus TTLB (trigger support).

A synchronization procedure requires at least two common triggers to be registered by both acquisition devices. One trigger is sent just after the beginning of the experiment ("start trigger") and another trigger is sent just before the end of the experiment ("end trigger"), and both are recorded by acquisition devices ACQD1 and ACQD2. A time reference is needed for synchronization purposes. This time reference can be provided either by one of the acquisition devices or by the master computer. Once the reference device has been selected, the other devices are called "secondary devices". The timestamps of the triggers of each secondary device are then linearly interpolated between the "start trigger" and the "end trigger" to align with the timestamps of the triggers from the reference device.

If we consider that N_T triggers were sent throughout the whole experiment, the formula applied for the transformation of the sample index of the trigger is:

$$s_{j1}^{(k)} = \left(s_j^{(k)} - s_j^{(1)}\right) \frac{s_1^{(N_T)} - s_1^{(1)}}{s_j^{(N_T)} - s_j^{(1)}} + s_1^{(1)} \quad (1)$$

where $\in \{2, 3, \dots, N_D\}$ } is the index of the secondary acquisition devices for a set-up with N_D devices (index 1 is reserved for the reference device), and $k \in \{1, 2, \dots, N_T\}$ is the trigger index with N_T the number of common triggers detected by the acquisition devices. Then, for the reference device and the j^{th} device, $s_1^{(k)}$ and $s_j^{(k)}$ are the trigger indexes of the k^{th} trigger in each respective data acquisition flow expressed in number of samples. In this context, $s_{j1}^{(k)}$ represents the new position of the k^{th} trigger from the j^{th} device after its alignment with $s_1^{(k)}$ in the data acquisition flow of the reference device (index 1). In this context, the alignment error $\varepsilon_s^{(k)}$ that affects the k^{th} trigger transformation, expressed in number of samples (subscript "$_s$") can be computed as:

$$\varepsilon_s^{(k)} = round\left(s_1^{(k)} - s_{j1}^{(k)}\right). \quad (2)$$

The errors $\left\{\varepsilon_s^{(k)},\ k = 2, \dots, N_T\right\}$ are theoretically bound by the ratio between the sampling periods of the acquisition devices. In *Figure 2A*, the sampling mechanism is schematized for an extreme case: an asynchronous trigger T (top line) appears slightly before the sampling time of the reference device (middle line), and slightly after the sampling time of a secondary device (bottom line). Thus, the maximum uncertainty, expressed in time (subscript "$_t$"), is $\varepsilon_t = \pm max(T_i)$ where T_i is the sampling period of the i^{th} acquisition device. By introducing R_T as the sampling ratio so that $R_T = max(T_i)/min(T_i)$, the maximum uncertainty can be rewritten as $\varepsilon_t = \pm R_T.min(T_i)$. For the faster device, the same error, expressed in terms of number of samples (ε_s) can then be written as $\varepsilon_s \in [-R_T, R_T]$.

To illustrate the performance of this synchronization procedure, we simulated ten thousand experiments of about one hour using two independent acquisition devices. Each device had its own time referential, its own nominal sampling frequency and its own random clock drift. With a 1‰ clock drift value being considered as usual, a random clock drift chosen in the interval $[-\frac{F}{1000}, \frac{F}{1000}]$ was added to the nominal sampling rate F of the corresponding acquisition device. Two situations were simulated. In the first one, the reference device had a sampling frequency ten times greater than the secondary device ($F1 = 10\ kHz, F2 = 1\ kHz, R_T = 10$). In the second case, both devices had the same sampling frequency ($F1 = 1\ kHz, F2 = 1\ kHz, R_T = 1$). The nominal sampling rates were affected by random clock drifts, as explained previously. For each simulation, the asynchronous triggers were added into recorded signals from both devices by an independent uniform random generator. On average there were $N_T = 3273$ asynchronous triggers per experiment. The transmission delay towards the acquisition devices was not

considered. The only uncertainty was, therefore, the date at which trigger signals were sampled according to the sampling clock of each acquisition device (*Figure 2A*).

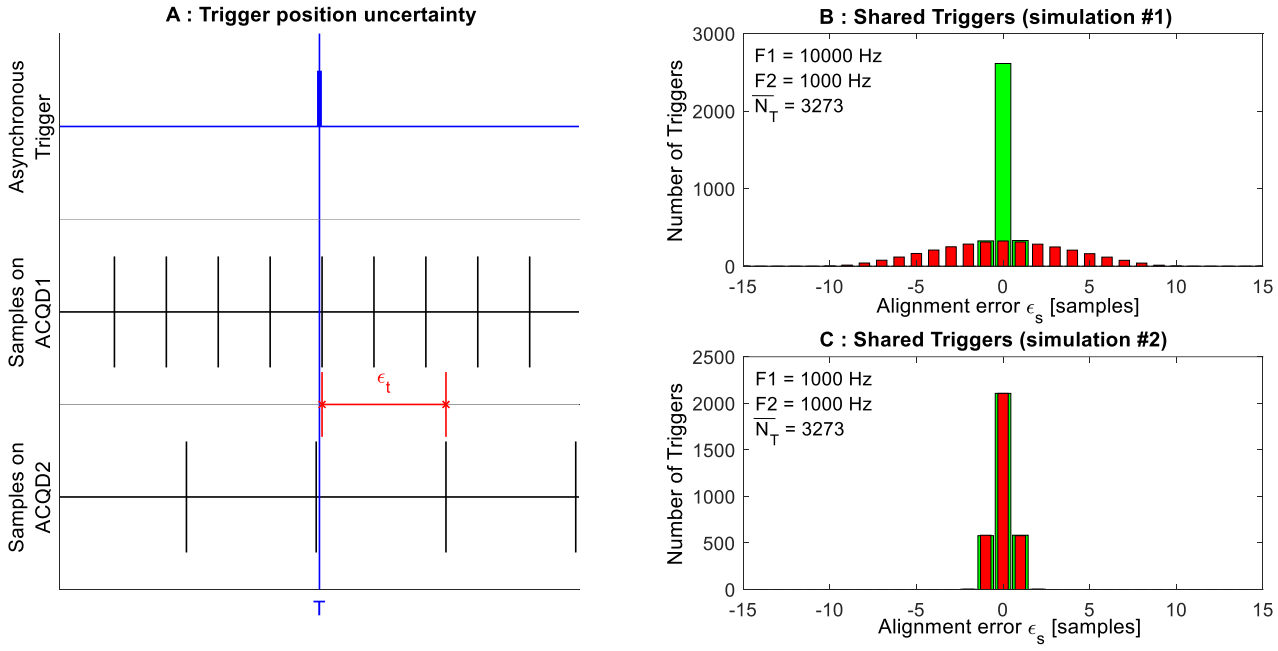

Figure 2. Trigger-based synchronization. A. Trigger uncertainty overview in an extreme case when the trigger T arrives just before and just after the sampling fronts. The vertical black segments represent the samples acquired by the ACQD1 and ACQD2 devices working at different sampling rates. B. Averaged histogram of alignment errors in samples (ε_s) with $R_T = 10$. C. Averaged histogram of alignment errors in samples (ε_s) with $R_T = 1,$. The red histogram illustrates errors from the fastest acquisition device compared to the slowest one, while the green histogram represents the reciprocal situation. Keys: $F1$ - ACQD1 sampling rate -; $F2$ - ACQD2 sampling rate -; $\overline{N_T}$ - average number of triggers detected by ACQD1 and ACQD2 -.

The range of the synchronization errors must be $\varepsilon_s \in [-10,10]$ in the first case, and $\varepsilon_s \in [-1,1]$ in the second case. These expected results are confirmed by the histograms of alignment errors (ε_s), averaged across all simulated experiments. These histograms are shown in *Figure 2B* ($R_T = 10$) and in *Figure 2C* ($R_T = 1$). The observed range is slightly larger when round-off errors are taken into consideration.

This simple synchronization procedure is an effective solution, but it must be based on three prerequisites: (1) all acquisition devices must be turned on before the first trigger is sent, and turned off

after the last trigger is sent, so that the start and end of the acquisitions are unambiguously specified by unique triggers; (2) all triggers must be sampled and shared by each acquisition device; and (3) the acquisition must be continuous, without any interruption, so that drifts are distributed linearly throughout the entire experiment.

In this ideal experimental situation, linear alignment of all triggers with the reference time is effective. However, because of handling errors, transient recording failures, false trigger detections and pauses during acquisition to cite some examples, an ideal situation is not always attained. Our procedure, which resolves these difficulties, is presented in the next section.

Description of the proposed synchronization procedure

The synchronization procedure proposed in this article efficiently corrects the clock drift even when the experimental situation is not ideal. The procedure allows correction when missing and/or spurious triggers are observed ("*Processing missing and spurious triggers*" section), and when interruptions in acquisition occur ("*Processing Pause/Resume mode*" section).

Processing missing and spurious triggers

The causes of missing or spurious triggers between different trigger streams are many and multifactorial. Triggers identified in the experimental design and sent by the "Stimulation PC" (*Figure 1*) but not recorded by all devices, are called "missing" triggers, as they are missing from at least one device. Triggers recorded by at least one device but not sent by the "Stimulation PC" are called spurious triggers. Configurations with missing triggers can be observed in a number of situations. If one device is switched on too late, or switched off too early, the first common trigger or the last common trigger determining the beginning or the end of acquisition is lost. This handling error can also result in a loss of data, when data

and triggers recorded on one acquisition device are not recorded on the others. Transient recording failure can have the same consequence. For example, a "buffer overflow" can occur in one system while the others continue recording. Configurations with spurious triggers can be observed in a number of situations. The detection of triggers is based on the electronic device that samples the signals on the parallel bus, and on the configuration mode of the parallel port. The name of a trigger is based on the decimal value corresponding to these sampled binary digits. We observed that the occurrence of additional spurious triggers depended on the electronic device used to sample and lock the logic levels on the parallel port. For example, in the analyzed recordings (cf. below the "*Validation on real data*" section), three different EEG systems were used. For two of these, no spurious triggers were observed, and for the other, up to 20% of the total number of triggers observed were spurious (*Table 1*). The spurious triggers observed corresponded to transient states on one sampling period before the expected trigger. Because the rise times and fall times can differ, their occurrence depended on the up or down transitions of logic levels on the parallel bus from one value (the previous trigger) to another (the subsequent trigger). Consequently, these spurious triggers appear in one trigger flow for a given device but not in the other trigger flows and have to be ignored. The proposed procedure resolves all of these issues.

String comparison is a central operation in a number of situations such as: in the comparison of two DNA sequences or in gene identification searches in molecular biology; in spelling error correction programs aimed at finding the dictionary entry which most resembles a given word; in the detection of plagiarism, and in pattern recognition (Needleman & Wunsch, 1970; Pearson & Lipman, 1988; Altschul, Gish, Myers & Lipman, 1990). As mentioned previously, when trigger flows from different devices are being compared, the alignment procedure may be deficient in situations featuring missing or spurious triggers. Before beginning the synchronization procedure, the common triggers in both acquisition flows must be matched specifically. This allows the triggers that mark the beginning and the end of acquisition to

be defined in a unique way. The "Longest Common Subsequence" (LCS) algorithm is proposed here as a way of resolving this problem (see Bergroth, Hakonen & Raita (2000) for a review). The LCS of two sequences X and Y constitutes a common subsequence of maximal length, i.e. the maximum number of identical symbols in both sequences when the common order in which these symbols occur in both sequences is maintained. The traditional implementation of the LCS between two sequences X and Y is by dynamic programming with a computation time proportional to the product of the lengths of the two sequences. The output is the longest sub-sequence with common triggers between the two sequences; the position of the common triggers may not be contiguous.

Figure 3 illustrates two trigger sequences, based on real acquisitions, used as inputs to the LCS algorithm (a first sequence recorded by the reference device called the reference Trigger sequence, RT, and a second sequence recorded by a secondary device called the secondary Trigger sequence, ST), and a unique trigger sequence (ALL) as output of the LCS algorithm. Each trigger is identified by a name which is based on the value transmitted in 8 bits by the parallel port to the acquisition devices. The two input sequences were recorded during a short real acquisition session with only 32 triggers in the RT sequence. The algorithm, based only on the triggers' position in the sequence, allowed us to obtain the longest subsequence of common triggers (ALL). The first and the last of these common triggers, which mark the beginning and the end of the synchronization procedure respectively, are of particular importance to the subsequent application of drift correction using *equation* 1. In the end, all the non-common (spurious) triggers were removed (green trigger in the RT flow and red triggers in the ST flow), and the output subsequence contained only the common triggers which were matched by the LCS algorithm, in which the first and last triggers of these subsequences were the "start trigger" (120) and the "end trigger" (96) respectively.

#	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
RT	100	120	201	110	10	11		161		90	94	110	10	11		165	90	95	110	10	11		166		90	95	110	10	11	164	90	96
ST		120	201	110	10	11	171	161	251	90	94	110	10	11	175	165	90	95	110	10	11	175	166	254	90	95	110	10	11	164	90	96
ALL		120	201	110	10	11		161		90	94	110	10	11		165	90	95	110	10	11		166		90	95	110	10	11	164	90	96

Figure 3. Illustration of the "Longest Common Subsequence" algorithm. RT is the reference sequence of triggers from the reference device. ST is a secondary sequence of triggers from a secondary device. ALL is the longest common subsequence of triggers obtained at the output of the algorithm. The trigger in green is the trigger "missed" by the secondary acquisition device, and triggers in red correspond to the spurious triggers introduced by the secondary acquisition device.

Processing Pause/Resume mode

As already listed above, in some experimental situations pauses are needed during recording. Acquisition phases are therefore separated by pauses which ensure that synchronization errors do not have a theoretical bound situated between $[-R_T, R_T]$ (in practice, it is slightly more, so that round-off errors can be taken into account).

Most acquisition devices have a Pause/Resume function to allow acquisition to be turned off when the recording of the analog signals is not useful, significantly reducing the size of stored files, and consequently shortening data processing time. These pauses can either be scheduled between each part of the experiment to allow participants to rest, or when a technical problem occurs. However, this very useful function can also generate unwanted effects: for instance, if acquisition includes one or more pauses, clock drift will no longer be linear. Consequently, a unique linear regression to fit the alignment of trigger timestamps from beginning to end of acquisition is not efficient. Instead, a piecewise linear regression is required, and a *shift correction* is introduced, as will be explained below.

Figure 4 below represents a simulated acquisition, containing three acquisition phases (segments GA, $B'C'$ and $D'E'$), separated by two pauses (segments AB' and $C'D'$). In this figure, the drift between the two clocks has been deliberately oversized for clarity. The ideal regression line for the time conversion between the two devices is the segment GG' (solid black plain line) with a slope equal to the nominal

unitary ratio ($R_T = 1$) of the sampling rates. Assuming that the secondary device (ACQD2) is slightly slower than the reference device (ACQD1), the segment GE (dotted black line) is situated below the ideal GG' segment with a unitary slope ($R_T = 1$), and the gap between the slopes of these two segments expresses the amount of clock drift between the two acquisition devices.

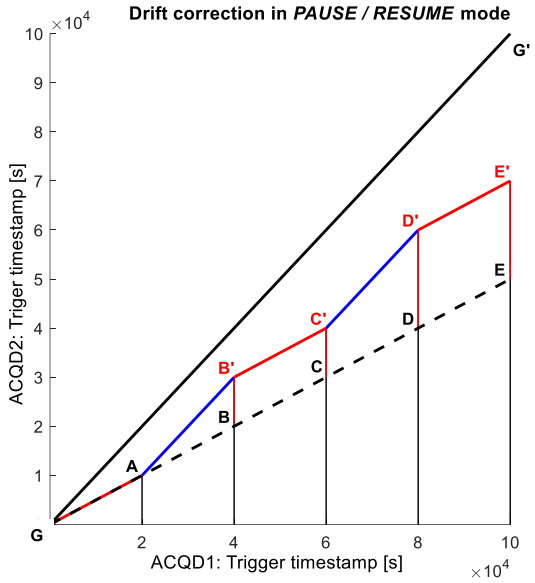

Figure 4. Clock drift correction when the Pause / Resume mode is used. Illustration of the theoretical alignment of the trigger timestamps [s] in a simulation with three acquisition phases (GA, $B'C'$, $D'E'$ in red) and two pause phases (AB', $C'D'$ in blue). The segment GG' corresponds to the ideal regression line, with $F_1 = F_2$, and the segment GE below illustrates the clock drift between the two devices when pauses are not taken into consideration.

Our reconstruction procedure is based on two assumptions. Firstly, during the acquisition phases, the timestamps of samples and triggers are linearly distributed following the slope of the segment GE. Consequently, segments GA, $B'C'$ and $D'E'$ representing the acquisition phases are parallel to the segment GE. Secondly, during the pause phases, only the duration of the pause and the nominal sampling rate (R_T) of the acquisition devices are known. Because synchronization is carried out off-line, the duration of each pause is easily estimated from the timestamp of the last sample before each pause and the timestamp of the first sample after each pause. Consequently, segments AB' and $C'D'$ representing the pause phases are

parallel to segment GG'. Thus, the principle of the supplementary mechanism of *shift* correction is as follows:

- The segments representing the acquisition phases after the first pause (i.e., $B'C'$ and $D'E'$) should be translated vertically to match the global acquisition segment GE.
- If the secondary acquisition device is slower (as it is the case in the illustration), a number of samples corresponding to segments BB' and DD' should be subtracted to correct this effect (conversely a number of samples should be added if the secondary device is faster).

Consequently, the amount of shift correction (length of BB' and DD') does not depend on the pause position in the acquisition session but is only proportional to pause durations.

A one-hour acquisition was simulated, with three acquisition phases, separated by two pauses with position and duration randomly chosen (147 s and 230 s respectively). In this simulation, there were 3252 common triggers with 1071, 1085 and 1096 triggers respectively for the first, second and third acquisition phases. The acquisition devices had the same nominal sampling rate (1 kHz, $R_T = 1$) and were affected individually by a random clock drift of under 1‰ ($F1 = 1000.114$ Hz for the first device and $F2 = 999.869$ Hz for the second one). By applying a unique alignment of the timestamps, the histogram of alignment errors (ε_s) was multimodal (*Figure 5A*) with three modes corresponding to the three acquisition phases. To compensate for the gaps after each pause, the 2nd and 3rd acquisition segments must be shifted by a certain number of samples, which have to be estimated by the shift correction procedure. In this example, the clock drift between the two devices was $F1 - F2 = 1000.114 - 999.869 = 0.245$ Hz. This would theoretically provide a shift of $0.245 \times 147 = 36.015$, rounded down to 36 samples after the first pause (duration 147 s), and a shift of $0.245 \times (147 + 230) = 92.365$ rounded down to 92 samples

after the second pause (duration 230 s). These results are supported by the fact that these shift values (36 samples and 92 samples) corresponded exactly to the shift between the modes' positions in the histogram in *Figure 5*A: $\Delta C_1 = 36$ samples and $\Delta C_1 + \Delta C_2 = 92$ samples.

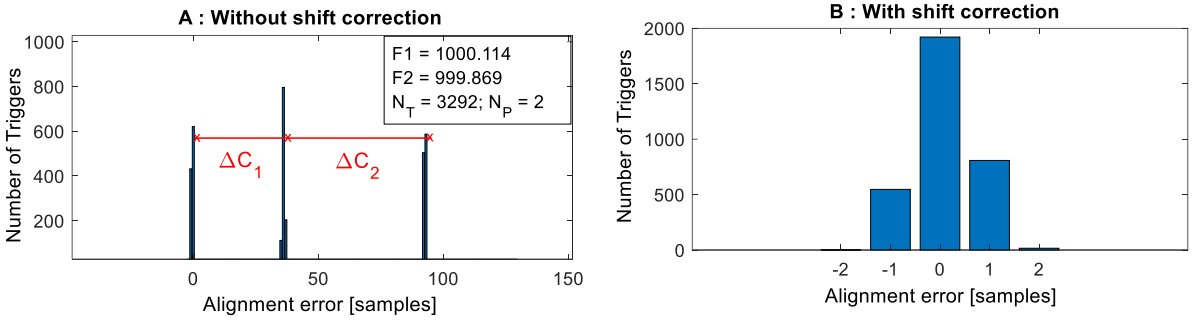

Figure 5. A. Histogram of alignment errors (ε_s) before shift correction. B. Histogram of alignment errors (ε_s) after shift correction. Key: $F1$ - ACQD1 sampling rate-; $F2$ - ACQD2 sampling rate-; N_T - number of triggers-; N_P - number of pauses-.

We will now describe in greater detail the implemented algorithm for the drift and shift corrections linked together. The regression line for the first acquisition phase is represented by the segment GA. For the subsequent acquisition phase, the value of the correction for the pause shift is found iteratively to vertically shift the acquisition phase (segment $B'C'$ for the second acquisition, and then segment $D'E'$ for the third acquisition phase) toward the real regression line. Two objective criteria stop this iterative process. The first one is the value of the coefficient of determination R^2 of the regression. It reaches its maximum when the acquisition segments are aligned. The second criterion is derived from the shape of the distribution of alignment errors. This means that the corresponding histogram should be as narrow as possible (not more than $2.R_T + 1$ bins), as symmetrical as possible, and centered on bin 0. The value of the correction is found at the end of the iterations. This correction value shifts all the trigger indexes belonging to this second segment of acquisition. This process is repeated for the third acquisition phase to estimate the second correction value, and all the way through to the last acquisition phase. After shift correction, a last linear

regression is performed, and the slope of the final regression line corresponds to the average slope of segments GA, $B'C'$ and $D'E'$. Assessment of the procedure is carried out from the final coefficient of determination R^2 and from the observed histogram of alignment errors.

The coefficient of determination R^2 was very close to 1 and the estimated value of the shift correction after the first pause was 43 samples (exactly the expected value). After the second pause, the estimated value of the shift correction was 110 samples (the expected value was 109 samples). The positions of the triggers for the secondary device were then linearly aligned on the time positions given by the sampling of the reference device. This was confirmed by the histogram of alignment errors (*Figure 5*B), which was, at this point, mainly described by three bins (positions -1, 0, 1). The theoretical bounds were $[-R_T, R_T]$ given $[-1,1]$ for $R_T = 1$. We also observed non-null values in the extreme bins of the histogram (positions -2 and +2). These values were due to the unavoidable jitters of the sampling rates and to the round-off errors.

In short, the proposed synchronization is composed of three steps: (i) the LCS algorithm, as described in the "*Processing missing and spurious triggers*" section, used to obtain the sequence of all common triggers detected by the acquisition devices, (ii) the clock drift correction with a piecewise linear regression for temporal alignment during each acquisition segment and the shift estimation for each pause phase, and finally (iii) a linear regression for complete temporal alignment after the application of shift corrections.

Validation on real data

In order to verify the effectiveness of our synchronization procedure, we tested it on real data (see "Datasets" section below). The assessment of its quality is illustrated in the "Synchronization procedure

quality assessment" section, and the comparison with another available implementation (EYE-EEG toolbox) in the "Comparison with the EYE-EEG toolbox" section.

Datasets

Data came from three different experiments (Frey, Ionescu, Lemaire, Lopez-Orozco, Baccino & Guérin-Dugué, 2013; Devillez, Guyader & Guérin-Dugué, 2015; Van Humbeeck, Meghanathan, Wagemans, Leeuwen & Nikolaev, 2018). The first two datasets came from our own records, and the third was provided by another laboratory, in order to validate our method using an independent dataset.

All three experimental designs used an eye-tracker (ACQD1) and EEG system (ACQD2) with the same setups shown in Figure 1. The eye-tracker was the same in all experiments (Eyelink 1000; SR Research), with a 1000 Hz sampling rate for the first two, and 250 Hz for the third. EEG signals were sampled at 1000Hz in the first experiment (32-channel BrainAmp™ system, Brain Products GmbH), at 1200 Hz in the second experiment (32-channel GAMMAsys gtec system, G.tec, Inc.), and at 250 Hz in the third (256-channel Electrical Geodesics System, Electrical Geodesics Inc., Eugene, OR). The eye-tracker was the reference device for the synchronization procedure, and the EEG system was the secondary device.

In Frey et al. (2013), nineteen participants took part in the experiment, with 180 trials per participant. For nine participants, the whole acquisition phase was carried out without using the Pause/Resume mode (one acquisition segment). For eight participants, the Pause/Resume mode was used once (providing two acquisition segments), and for two participants, there were two pauses (three acquisition segments). These three situations are synthesized in *Table 1* and labeled scenarios 1.1, 1.2 and 1.3 respectively. The Devillez et al. (2015) experiment consisted of a visual search experiment, with thirty-nine participants and two conditions, and sixty trials per condition. Among the seventy-eight available datasets, forty-eight (labeled scenario 2 in Table 1) were randomly selected in order to have the same order

of magnitude of the total number of triggers recorded as with scenario 1 datasets. Finally, one dataset was extracted from the third experiment (Van Humbeeck et al., 2018), in which twenty-three participants carried out a contour integration task. This dataset appears in *Table 1*, and is entitled scenario 3. In scenarios 2 and 3, the Pause/Resume mode was not used and there was only one acquisition segment per recording.

Scenario	1.1	1.2	1.3	2	3
EEG system	BrainAmp	BrainAmp	BrainAmp	GTec	EGI
EEG sampling rate [Hz]	1000	1000	1000	1200	250
ET sampling rate [Hz]	1000	1000	1000	1000	250
# Datasets	9	8	2	48	1
# Segments (# Pauses)	1(0)	2(1)	3(2)	1(0)	1(0)
# Common triggers	11235	9916	2413	26578	482
# Missing triggers	0 (0%)	0 (0%)	0 (0%)	0 (0%)	968 (66.8%)
# Spurious triggers	2678 (19.2%)	2359 (19.2%)	622 (20.5%)	0 (0%)	0 (0%)

Table 1. Dataset description. Keys: The five different datasets are labeled scenarios 1.1, 1.2, 1.3, 2 and 3. Each is described with the type of EEG system, the EEG sampling rate, the eye-tracker (ET) sampling rate, the number of datasets, the number of acquisition segments, the number of pauses, the cumulative number of the common triggers for all datasets, of the missing triggers and of the spurious triggers detected in the EEG data flow. The percentages in parentheses were computed in relation to the total number of triggers.

These datasets enabled us to assess the quality of the proposed synchronization procedure in various situations for $R_T = 1$ (scenarios 1.1, 1.2, 1.3 and 3) and $R_T = 1000/1200$ (scenario 2), including the problem of spurious triggers (scenarios 1.1, 1.2, and 1.3), the problem of missing triggers (scenario 3) and the use of the Pause/Resume mode (scenarios 1.2 and 1.3). A dataset illustrating each of these scenarios is

available (Zenodo repository: https://doi.org/10.5281/zenodo.5554071). The datasets were synchronized using our software, named “GazeEEGSynchro”, which implements our synchronization proposal and which is also available in the Zenodo repository. A tutorial showing how to use the “GazeEEGSynchro” software is provided in the “Supplementary Material”. After synchronization, data from the EEG and eye-tracker recordings were grouped together, and as such became the synchronized data. The synchronized data came from the EEG channels, the vertical and horizontal eye positions, and a “blink” logic signal identifying the onset and the offset of each blink. When the data from the eye tracker were added to the EEG channels from EEG recording, there were, therefore, either three extra channels when recording of ocular data was in monocular mode, and five extra channels when recording of ocular data was in binocular mode.

Synchronization procedure quality assessment

In the following sections, we evaluate the quality of our synchronization procedure in two ways. The first section, "Trigger alignment", is based on the distribution of the alignment errors. The second section, "Blink alignment", is based on blinks, which are easily detected on the eye-tracker, and on EEG signals. This was done to estimate the time delay between blink onsets from the two modalities, throughout the experiment, with and without clock drift correction. Two implementations were performed, one on all scenario 1 datasets, and the other on all scenario 2 datasets. In the latter case, an additional step of resampling the synchronized data was required, as the two devices did not initially have the same sampling rate.

Trigger alignment

In this section we present the distribution of the alignment errors on the trigger timestamps after clock drift correction from acquisitions using the Pause/Resume mode (scenarios 1.2 and 1.3). To do so, we selected

two random datasets, the first from scenario 1.2 with one pause (two acquisition segments), and the second from scenario 1.3 with two pauses (three acquisition segments). The synchronization was applied separately to the two datasets. We expected to observe multimodal distributions of the alignment errors when clock drift correction was applied overall, on the complete acquisition, and monomodal distributions when the shift correction was incorporated into the clock drift correction.

Figure 6A presents the observed histograms of alignment errors for the first dataset, without shift correction, i.e. when the clock drift correction was applied once on the complete acquisition. The number of alignment errors in this histogram represents the number of common triggers after application of the LCS algorithm. There were 1304 common triggers distributed over two acquisition phases, respectively 271 for the first phase, and 1033 for the second. As explained previously ("*Processing Pause/Resume mode*" section), this histogram is composed of two modes corresponding to the two acquisition phases separated by one pause phase. *Figure 6B* presents the observed histogram of alignment errors, with a shift correction included into the clock drift correction.

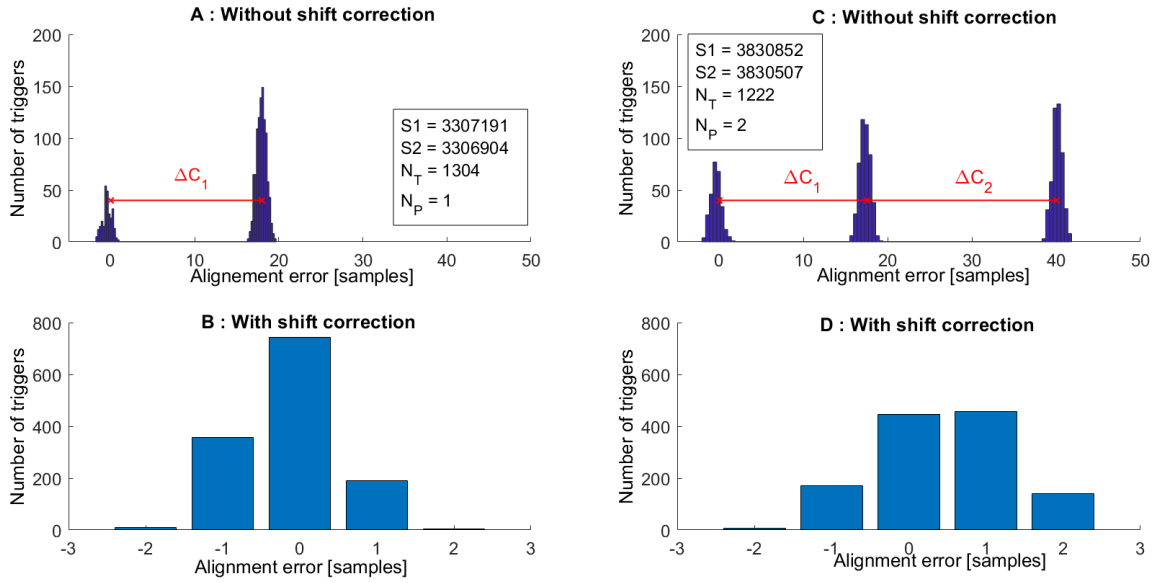

Figure 6. Alignment errors on real data from acquisitions using the Pause/Resume mode, (left plots scenario 1.2; right plots scenario 1.3). A. Histogram of alignment errors (ε_s) without shift correction (one pause, two acquisition phases). B. Histogram of alignment errors (ε_s) with shift correction (one pause). C. Histogram of alignment errors (ε_s) without shift correction (two pauses, three acquisition phases). D. Histogram of alignment errors (ε_s) with shift correction (two pauses). Key: S1 - number of eye-tracker samples-; S2 - number of EEG samples-;N_T - number of common triggers-; N_P - number of pauses-.

Similarly, for the second dataset, the graph presented in *Figure 6*C illustrates the observed histogram of alignment errors after clock drift correction but without shift correction. For this dataset, there were 1222 common triggers distributed as follows: 273 triggers during the first acquisition phase, 469 during the second one and 480 during the third one. This histogram is composed of three modes, one for each acquisition phase. See *Figure A1* in the Appendix for the joint representation of the trigger timestamps in both modalities showing the three acquisition phases and the time shifts between them. *Figure 6D* presents the observed histogram of alignment errors with a shift correction included into the clock drift correction. The shift value for each pause was estimated by the shift correction procedure as explained in the "*Processing Pause/Resume mode*" section. For the first dataset with one pause, the estimated applied shift was equal to 17 samples. As *Figure 6A* shows, this value corresponds to the shift ΔC_1 between the two modes. For the second dataset, with two pauses, the estimated applied shift for the second acquisition segment was equal to 17 samples, and for the third acquisition segment, a last estimated shift was equal to 22. As *Figure 6C* shows, these two values were in line with the shifts observed between the histogram modes (ΔC_1, ΔC_2). Finally, after clock drift correction with shift correction, the R^2 coefficient was very close to one 1 for the final linear regression. This meant that the trigger positions of the reference and secondary devices were very well aligned in each synchronized dataset. This was confirmed by the histogram of alignment errors after clock drift correction and with shift correction (*Figure 6B*, *Figure 6D*), where the non-null values of the extreme bins of the histogram (position ± 2) were due to the unavoidable jitters of the sampling periods and to errors due to rounding.

512

513 **Blink alignment**

514 In order to assess the overall quality of our procedure throughout the entire experiment, we analyzed
515 the temporal delay between common events which were almost simultaneous and easily detectable in both
516 modalities. Using the notation presented in the "Principle of trigger-based synchronization" section, $t(s_1^{(i)})$
517 and $t\left(s_2^{(i)}\right)$ represent the onset of the ith event extracted from the dataset of the first acquisition device (the
518 reference device), and from the dataset of the second acquisition device, respectively. The difference
519 $t(s_1^{(i)}) - t(s_2^{(i)})$ is the temporal delay for the onset of the same event ith in both acquisitions. After
520 synchronization, $t\left(s_{21}^{(i)}\right)$ represents the onset of the ith event extracted from data recorded by the second
521 acquisition device in the synchronized dataset using the first device as reference. After synchronization
522 $t(s_1^{(i)})$ becomes the reference time for the onset of the ith event. The difference $t(s_1^{(i)}) - t(s_{21}^{(i)})$ represents
523 the temporal delay for the onset of the same ith event in the synchronized dataset. The former difference is
524 denoted by $\Delta t_{NO}\left(s^{(i)}\right)$, which corresponds to the delay without clock drift correction represented by
525 $\Delta t_{YES}\left(s^{(i)}\right)$, which corresponds to the delay with clock drift correction. Without clock drift correction, the
526 delay $\left|\Delta t_{NO}\left(s^{(i)}\right)\right|$ in absolute value should increase along acquisition as the accumulated amount of clock
527 drift. However, since the event is assumed to be almost simultaneous in both modalities, when clock drift
528 correction is applied, the delay $\Delta t_{YES}\left(s^{(i)}\right)$ should be relatively small, and in the same order of magnitude
529 as alignment errors.

530 Eye blinks were the ideal candidate for this "almost simultaneous and easily detectable common
531 events" in both modalities. Blinks are generally involuntary acts of shutting and opening the eyelids, and
532 this interrupts the saccade-fixation series. Both blinks and saccades can mask brain activity, because they
533 elicit larger electric potentials than baseline signals (Picton et al., 2000). For our purposes, we used the

blinks detected by the eye-tracker as a reference, and those estimated from the EEG signals, throughout the experiment. The time delays between the blink onset in each modality, with and without clock drift correction ($\Delta t_{YES}(.)$, $\Delta t_{NO}(.)$) were computed. For all datasets from scenarios 1 and 2, and without clock drift correction, the two trigger flows were only aligned on the same first common trigger to set the same timestamp (0) for the beginning of recording in both modalities. After that, the synchronization with the complete clock drift correction was applied from the first common trigger up to the last common trigger. This synchronization included a resampling step at 1000 Hz for EEG signals from scenario 2 (initial EEG sampling rate at 1200 Hz, see *Table 1*) in order to obtain the same sampling rate as the reference device (eye-tracker at 1000 Hz). Resampling was not performed on datasets from scenario1 because the two devices had the same sampling rate of 1000 Hz (see *Table 1*). The scenario 3 was not used for this assessment because there was only one dataset and the configuration with the same sampling rate was already considered with scenario 1 datasets.

The logic Blink signal (True during blink, False otherwise) was used to time-lock the EEG signal on the FP1 channel which was epoched within a temporal window of $[-0.5;\ 0.5]$ seconds on each side of the blink onset. The number of epochs was therefore equal to the number of blinks. Blink duration was estimated from the duration of the logic Blink signal, and only epochs with blinks lasting between 100 and 400 ms were retained for analysis: 2507 epochs from scenario 1 and 4919 from scenario 2, which on average corresponded, respectively to 132 and 102 epochs per participant. On the EEG signal, blinks were detected on the FP1 channel (Jung, Makeig, Westerfield, Townsend, Courchesne & Sejnowski, 2000), and were related to the local maximum value of the first derivative, computed with a Canny-Deriche Filter (Deriche, 1990).The parameter α of the filter was calculated so that: $\alpha \cdot \sigma = 2.5/\sqrt{\pi}$ where σ is the width of the Gaussian filter. To ensure accurate detection, the following values were empirically set: $\sigma = 10$ and $\alpha = 0.14$.

Figure 7 illustrates five epochs (index 2, 30, 60, 89, 118) distributed across the experiment (towards the beginning for the first epochs, and towards the end for the last epochs), for a given participant. The first two dotted lines represent the FP1 signal and its derivative without clock drift correction. The Blink signal is represented on the third line. The fourth and the fifth plain lines represent the same FP1 signal and its derivative after clock drift correction.

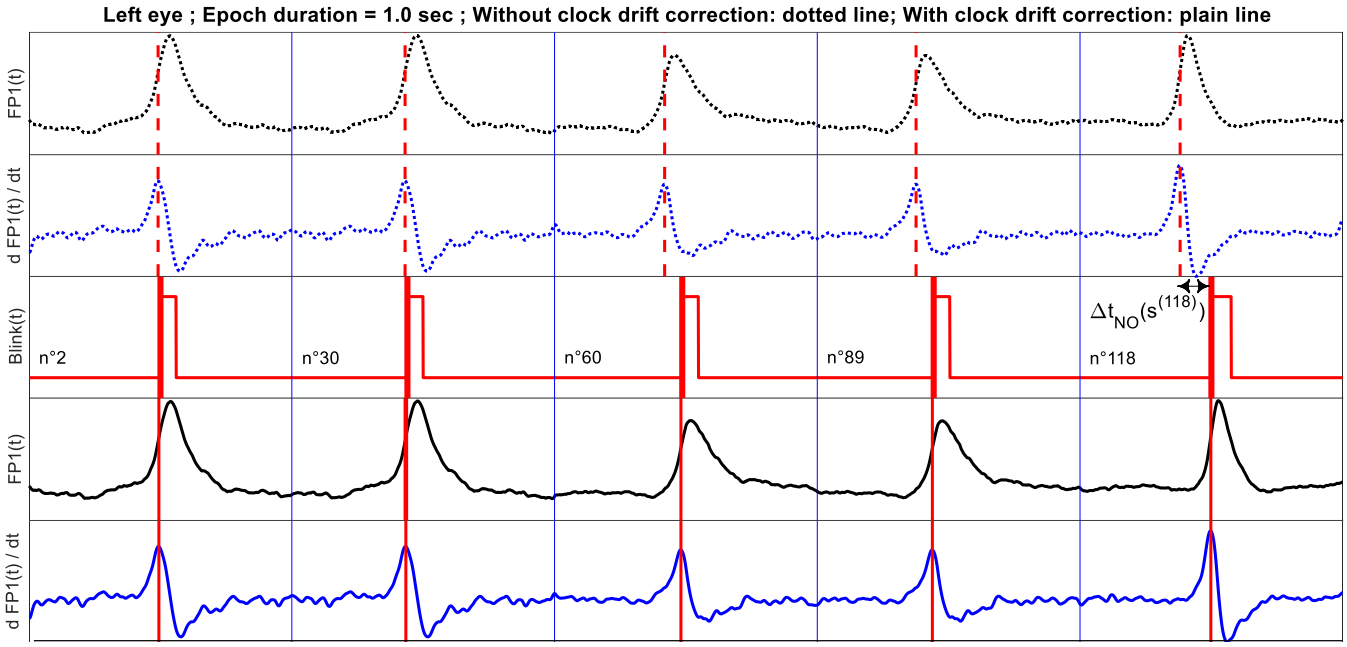

Figure 7. Illustration of the clock drift correction observed on eye blink events. The vertical blue lines delimit the end of each epoch. Selected channel: FP1 from EEG data and left eye blink from eye-tracker data. Upper two signals: channel FP1 and its first derivative without clock drift correction. Middle signal: logic Blink signal. Lower two signals: channel FP1 and its first derivative after clock drift correction. The vertical dashed red markers indicate blink detection on the EEG signal ($FP1(t)$) from the first derivative $d\,FP1(t)/dt$ on the dotted line without clock drift correction. For epoch number 118 (i.e. the 188th blink), $\Delta t_{NO}\left(s^{(118)}\right)$ is the delay for the 188th blink onset between both modalities without clock drift correction. The vertical plain red markers indicate blink detection on the EEG signal ($FP1(t)$) from the first derivative $d\,FP1(t)/dt$ in plain line with clock drift correction.

In Figure 7, in the absence of correction, clock drift is clearly visible from the increasing length of delay between the blinks detected by the eye-tracker (vertical plain red line third on the Blink signal) and

the blinks detected on the EEG signal (the vertical dashed red line indicates the maximum value of the first derivative on the dotted line). These delays are represented by $\Delta t_{NO}(s^{(2)})$, $\Delta t_{NO}(s^{(30)})$, $\Delta t_{NO}(s^{(60)})$, $\Delta t_{NO}(s^{(89)})$, $\Delta t_{NO}(s^{(118)})$[5] and qualitatively $\Delta t_{NO}(s^{(2)}) < \Delta t_{NO}(s^{(30)}) < \Delta t_{NO}(s^{(60)}) < \Delta t_{NO}(s^{(89)}) < \Delta t_{NO}(s^{(118)})$. However, when clock drift correction is applied, these delays (i.e., delay between the vertical plain red lines on the Blink signal, and the vertical plain red line indicating the maximum value of the first derivative in plain line) are so short that they are not visible in *Figure 7*.

In order to quantify these delays in scenarios 1 and 2, without and with clock drift correction, we computed the delays $\Delta t_{N0}(s^{(i)})$ and $\Delta t_{YES}(s^{(i)})$ for all epochs. In this way, each blink was characterized by three features (all in ms): (1) its absolute timestamp in the experiment, i.e. its onset given by the eye-tracker $t(s_1^{(i)})$ as the reference, (2) the corresponding delay $\Delta_{NO}(s^{(i)})$ without clock drift correction, and (3) the corresponding delay $\Delta_{YES}(s^{(i)})$ with clock drift correction. *Figure 8* left shows the joint distributions of Δ_{NO}vs t (red dots) and Δ_{YES} vs t (green dots) for data from scenario 1 (Top) and for data from scenario 2 (Bottom).

[5] Because this delay is quite long, it is visible in *Figure 7*.

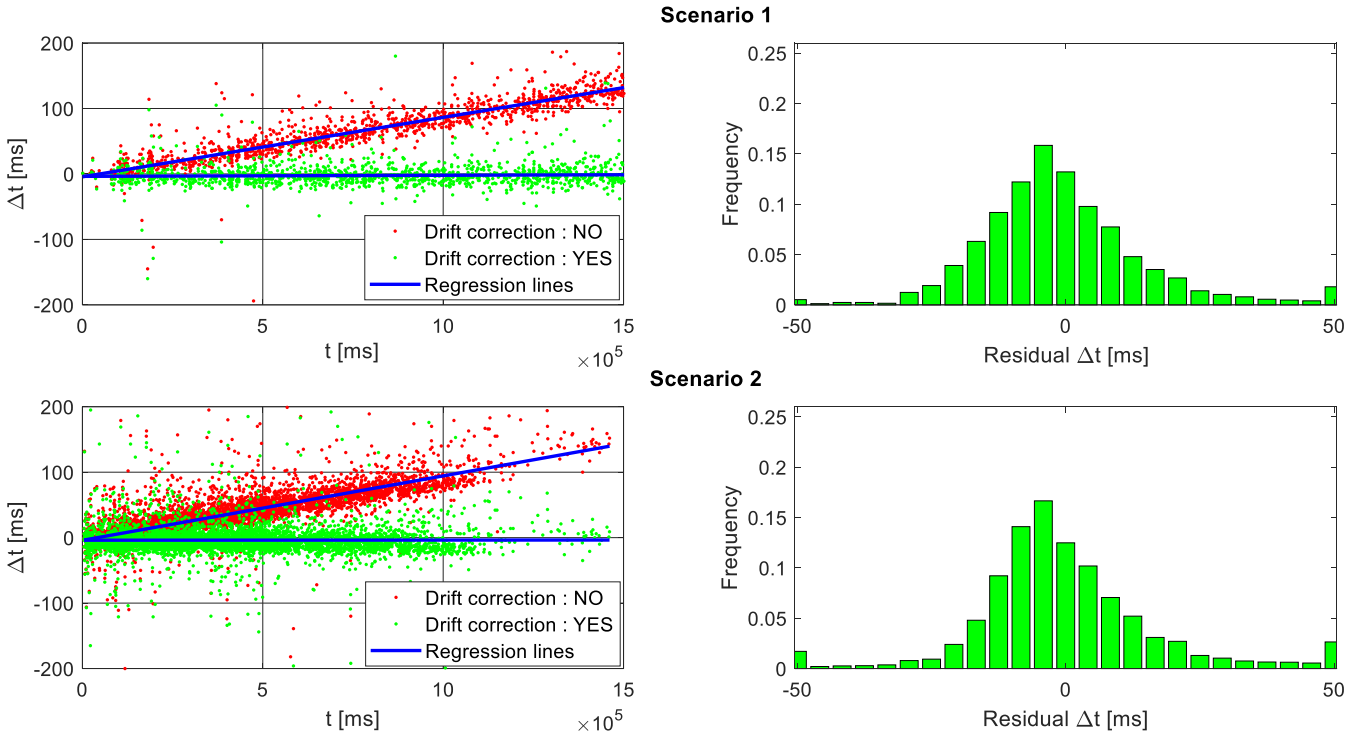

Figure 8. Statistical analysis of blink detection delay. Left: joint representation of the eye blinks collected from scenario 1 (Top) and from scenario 2 (Bottom). The red dots represent the Δ_{NO} data (without drift correction) while the green dots represent the Δ_{YES} data (with the drift correction procedure). Time on the horizontal axis is expressed in milliseconds. Right: Histogram of the residual errors between Δ_{YES} and the values predicted by the linear regression, from scenario 1 (Top) and from scenario 2 (Bottom). The extreme bins of the histograms correspond to "less than -50 ms" and "more than 50 ms".

As expected, and plotted separately for the two scenarios on *Figure 8* left, a linear regression model attempts to fit the linear relationship between the Δ_{N0} or Δ_{YES} delays as a function of the absolute timestamp t of the blink onset. The two regression lines are represented in blue on Figure 8 left, and their slope (a) and intercept (b) are given in Table 2. *Figure 8* right shows the histograms of residual errors, i.e., the differences between the observed values of delay Δ_{YES} and the values predicted by the linear regression line. The mean and standard deviation (in brackets) of these distributions were -1.34 (39.83) ms for the first scenario, and -3.83 (35.07) ms for the second one.

	Epoch number	Δ_{NO} Without clock drift correction	Δ_{YES} With clock drift correction
Scenario 1	23564	a = $8.8901.10^{-5}$ b = 0.1291 ms	a = $3.3520.10^{-7}$ b = 0.2268 ms
Scenario 2	26578	a = $9.5372.10^{-5}$ b = 0.8523 ms	a = -2.265210^{-6} b = 0.7970 ms

Table 2. Slope (a) and intercept (b) of the linear regression between the dependent variable (Δ_{NO} or Δ_{YES}) and the independent variable t representing the blink onset given by the eye-tracker.

We observed similar values for the slopes (a) in scenarios 1 and 2, whereas the synchronization for the latter included a supplementary resampling step at 1000 Hz for EEG signals. More importantly, we observed a decrease of two orders of magnitude between the slopes (a) without and with drift correction. Interpretation of slope a seems straightforward. In both scenarios, for example, the two slopes were $8.8901.10^{-5}$ and $9.5372.10^{-5}$ ie. about 10.10^{-5}, when clock drift was not corrected. With such a value, and for an acquisition time of 15.10^{5} ms, the synchronization delay is about 150 ms (*Figure 7*). Thus, for a one-hour experiment (36.10^{5} ms), the progressive desynchronization time between the two data flows would come to 360 ms at the end of the experiment. This shift value is clearly not negligible, and in the context of our EFRP data, could lead to overlapping and/or misinterpretation of the resulting EEG components. After synchronization, a slope of around 10^{-6} can be expected. Consequently in a one-hour experiment, progressive residual desynchronization would be reduced by 100 (around 3.6 ms), which corresponds to a negligible number of samples.

Comparison with the EYE-EEG toolbox

We applied our method implemented in GazeEEGSynchro software, and the synchronization method implemented in the EYE-EEG environment (Dimigen et al., 2011), to datasets from scenarios 2 and 3 in order to compare their respective main functionalities, overall quality and time processing capacities. Indeed, the scenario 2 had the largest number of datasets.

Input/Output information

Both procedures require the eye-tracker and the EEG files as inputs. However, for the EYE-EEG method, the names of the two common synchronization triggers have to be entered, one indicating the beginning of acquisition, and the other the end. These inputs are not necessary with our solution because of the alignment of triggers provided by the LCS algorithm (cf. "*Processing missing and spurious triggers*" section).

Regarding output, the interface can be used to select which ocular channel needs to be added with the EYE-EEG method. With our implementation, all ocular channels are added, and after synchronization, users can remove unnecessary channels as required.

In the EYE-EEG method, the reference device is always the EEG device. This means that synchronized data and triggers are sampled at the same frequency as the EEG device. As a result, only the ocular channels are interpolated on the EEG sampling timestamps. With our implementation, users can choose the reference device.

Common triggers: selection and regression

In our method, common triggers are automatically detected by the LCS algorithm. The start and end triggers are directly identified as the first and last common triggers, respectively. Only data from EEG and

644 eye-tracker recordings between these two timestamps are synchronized, either on the EEG timeline or on
645 the eye-tracker timeline, depending the user's choice of reference device.
646 In the EYE-EEG method, users have to provide the start and end triggers. Common triggers are then
647 searched for between these two triggers, and are determined in two steps. The first step is the intersection
648 of the trigger lists in each modality. Spurious triggers and missing triggers are therefore naturally excluded
649 from this common list. The second step consists of finding which of these triggers can be matched in both
650 modalities according to their timestamps. The eye-tracker timestamps of the start and end triggers are
651 arranged linearly to match with the corresponding triggers in the EEG timeline. After this linear
652 transformation from the eye-tracker timestamps to EEG timestamps, the matching of each common trigger
653 is performed if the gap between the original timestamp on the EEG timeline and the transformed timestamp
654 is smaller than a given tolerance (four samples, default value of this parameter). One single linear
655 regression is performed on the timestamps of each common trigger in order to globally optimize the
656 timestamps of the common triggers and to find the linear relationship between the EEG timeline and the
657 eye-tracker timeline.
658 We have shown that a piecewise linear regression needs to be carried out for acquisition in
659 Pause/Resume mode, and this functionality is not implemented in the EYE-EEG toolbox. Consequently,
660 synchronization of a scenario 1.3 dataset containing two pauses (*Table 1*, *Figure 6* C D) was not successful
661 with the Eye-EEG method (*Figure A2* in the Appendix).
662 Finally, *Table 3* summarizes the main differences between both methods according to their
663 functionalities.

	Proposed synchronization method	EYE-EEG
Implementation	DOS application written in C++	Matlab® functions integrated into EEGLab

Eye-tracker	SR Research	SR Research, SMI, Tobii, Tobii Pro
EEG	Converted to BrainVision format[6]	EEG format supported by EEGLAB
Reference device	Eye-tracker or EEG	EEG
Ocular channels	All	Selected by user
Trigger identification: begin / end recording	Automatic	Selected by user
Recording with Pause/Resume mode	Yes	No
Evaluation from alignment errors		
- Alignment errors for all common triggers	In text format (log file)	Not directly provided, but can be computed from available variables
- Global histogram	In text format (log file)	
- Specific histogram for each trigger type	In text format (log file)	In stored variables and in graphic
- Overview of event latencies in output datasets	Not provided	Not provided Graphic
- Regression between event latencies	Coefficient of determination	Graphic and coefficient of determination
- Trigger and event counting	Provided	Provided
Evaluation from cross-correlation	Must be implemented by user in their preferred environment	In stored variables and in graphic

Table 3. Comparison of the main functionalities of both methods.

[6] Use converter functions provided by different environments (EEGLAB, BrainStorm, MNE-Python, etc.)

667 **Overall quality**

668 The quality assessment was performed on the scenario 3 dataset for illustration, and on scenario 2
669 datasets to provide statistical comparisons.

670 Each type of synchronization software provided a histogram of alignment errors after
671 synchronization, expressed in number of samples. Each histogram (*Figure 9*) was computed on 11 bins,
672 from -5 samples up to 5 samples. For each histogram, the following statistical features were computed:

673 - Position: mean (best when close to zero),
674 - Spread: standard deviation (best when close to zero),
675 - Flatness: Kurtosis coefficient (best when high),
676 - Asymmetry: Skewness coefficient (best when close to zero).

677

678 The EYE-EEG method provided better alignment errors for the scenario 3 dataset than our proposed
679 method, according to the statistical features computed on the distribution of alignment errors. The
680 corresponding histograms are illustrated in *Figure 9*: “position” closest to zero (0.035 vs 0.475), better
681 “spread” (0.317 vs 0.449), better “flatness” (9.663 vs 4.436) and better “asymmetry” (0.111 vs 1.058).

682

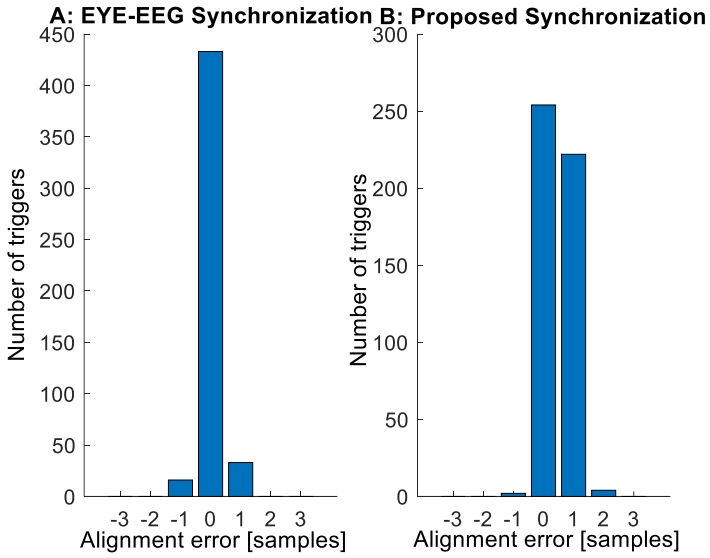

Figure 9. Histogram of alignment errors after synchronization for the scenario 3 dataset, for (A) the EYE-EEG synchronization and (B) the proposed synchronization. The counts for bins -5, -4, 4 and 5 were zero and were not represented.

Dimigen et al. (2011) first introduced cross-correlation to assess the quality of ocular artefact reduction in EEG signals, and of synchronization. For this later, the cross-correlation was computed between the horizontal position of the eye, and the difference between the amplitudes recorded on the right and left electro-oculogram (horizontal EOG) channels. Segments featuring blinks and signals that were too noisy were removed beforehand. The cross-correlation was computed with the "xcorr" function provided by the "Signal Processing" toolbox in Matlab®.

Figure 10 A B illustrates the two signals obtained from a randomly-selected trial, and *Figure 10* C D shows the cross-correlation result. The position of the maximum value of the cross correlation must be close to zero, and lag is defined by the time position of this maximum value. A positive lag means that the transition of the horizontal eye position occurs later than the transition in the EOG signals. For this dataset, the cross-correlation reached its maximal value for the same lag length (3 samples) in both methods.

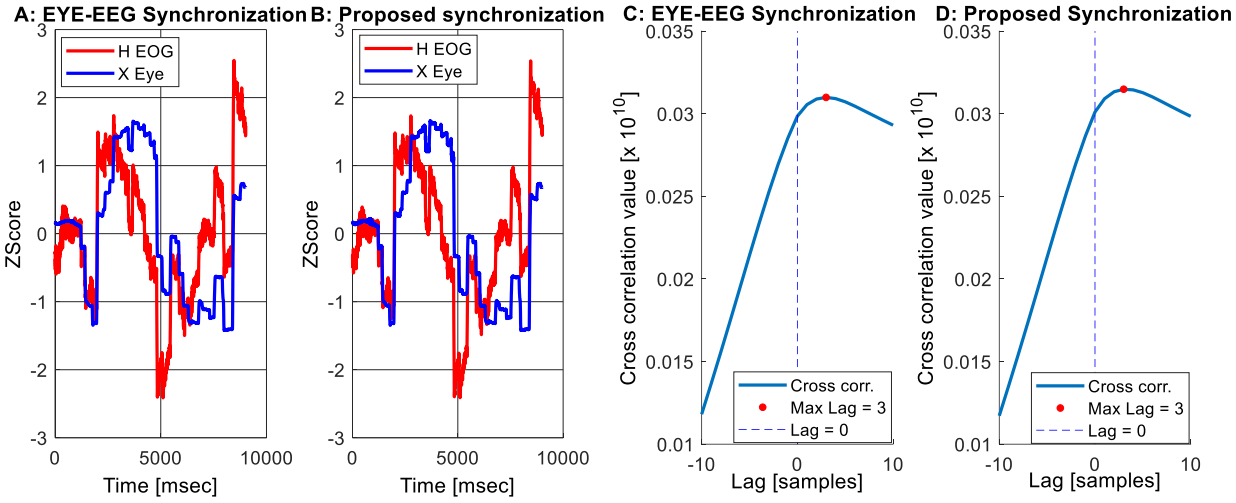

Figure 10: For the scenario 3 dataset, (A, B) evolution of the horizontal eye position and the horizontal EOG (H EOG) signals obtained from a randomly-selected trial, after centering and reduction (ZScore) to enable superposition on the same scale, (C, D) cross correlation between these two signals, for (A, C) the EYE-EEG synchronization and (B, D) the proposed synchronization

All these features based on alignment errors and cross-correlation were computed on scenario 2 datasets for each method of synchronization. *Table 4* shows the statistical summary for each feature and each implementation.

	Proposed synchronization method	EYE-EEG	Student test
Mean [samples]	-0.015 (0.062)	-0.020 (0.036)	t(47) = 0.030, p = 0.977
Standard dev. [samples]	0.551 (0.007)	0.699 (0.005)	t(47) = -4.220, **p <0.001** ***
Kurtosis	2.604 (0.024)	2.695 (0.030)	t(47) = 13.619, **p <0.001** ***
Skewness	0.110 (0.116)	-0.064 (0.050)	t(47) = 1.393, p = 0.170
Cross corr. lag [samples]	-0.875 (0.151)	-1.313 (0.196)	t(47) = 2.445, **p = 0.018***

Table 4. Statistical summary. Keys: average (standard error) of the quality features for both implementations (column 2, 3); results of the Student tests (column 4), for four statistical moments (Mean, Standard deviation, Kurtosis and Skewness) and the cross-correlation lag. *p<0.5, ***p<0.001, bold: significant effect.

Results for the spread feature (std) and the cross-correlation lag[7] were significantly better with our synchronization method, whereas the flatness (Kurtosis) was significantly better with the EYE-EEG synchronization method.

Processing time

On average, our implementation ran 7.5 times faster than the EYE-EEG synchronization for scenario 2 datasets, on the same computer. This result is mainly due to the fact that programing languages are different in both implementations (C++ for ours and Matlab® for EYE-EEG). More interestingly, *Table 5* shows the percentage of time spent on data loading compared to synchronization computation. In our implementation, most of the time was devoted to data loading and implementation of the synchronization algorithm was optimized, making it extremely fast.

	Eye tracking data [%]	EEG data [%]	Synchronization [%]
Proposed synchronization method	55.2	44.5	0.3
EYE-EEG	79.2	14.1	6.7

Table 5. Comparison of the percentage of time spent on data loading (eye-tracker data and EEG data) and on synchronization.

[7] EOG signals were not recorded for scenario 2 datasets. For cross-correlation estimation, EEG channels closest to the external canthus of the right/left eye were therefore chosen: F8 (right) and F7 (left) channels.

729

730 **Conclusion**

731 In this article, we present an original method of data synchronization, generated by acquisition
732 devices working independently. The core of the method rests on a classical synchronization paradigm using
733 external triggers. However, we have shown that this paradigm alone does not provide a sufficiently robust
734 synchronization when dealing with a wide range of problematic situations. We therefore proposed a
735 complete procedure, which we tested and compared with the open source EYE-EEG toolbox on simulated
736 data and on real data from co-registrations of eye movements and EEG.

737 Our procedure specifically aimed to compensate in situations with missing/spurious triggers in the
738 different data flows, and for use of the Pause/Resume mode by the experimenter during acquisition.
739 Common triggers marking the beginning and the end of acquisition must be accurately detected. If
740 synchronization is based only on the detection of these two common triggers, results can be inconsistent
741 when these triggers are missing, for example, if the experimenter starts the acquisition of the secondary
742 device too late or stops it too early. In such situations, the EYE-EEG toolbox requires a manual search in
743 raw files in order to choose other common triggers. Our procedure has the advantage of automatically
744 rectifying errors generated by the experimenter. In addition, if electronic circuits are relied on to sample
745 trigger signals, spurious triggers corresponding to transient states on one sampling period can appear and
746 render the matching process ambiguous. For this reason, we proposed the use of the LCS algorithm for the
747 matching of trigger sequences. We showed that this flexible matching algorithm is a very useful first step
748 before clock drift correction.

749 Moreover, as we have pointed out, there is a growing interest in co-registration experiments using
750 complementary experimental modalities. However, these experiments are often fairly long, and may require
751 pauses, which can either be pre-planned to allow participants to rest for example, or unplanned, when

technical problems occur on one or other of the recording devices. In both cases, the use of the "Pause/Resume" mode makes it possible to limit the size of the recorded files, and to facilitate subsequent processing on samples/files that contain only informative data. It is therefore important to have a synchronization procedure which takes these pauses into consideration and manages them correctly. To this end, we implemented a piecewise linear regression between the common triggers expressed in number of samples in each data flow. Drift correction was not, therefore, carried out across all trigger sequences and was instead done locally on each acquisition phase between the successive pauses. Shift values were automatically calculated by the algorithm to compensate for pauses. By executing our procedure on real data corresponding to an acquisition of around one hour, we observed very few samples of residual drift.

After synchronization, quality can be assessed from the distribution of alignment errors, and from cross-correlation between horizontal EOG and eye position. The spread of the distribution of alignment errors is on average larger when synchronization is carried out with the EYE-EEG toolbox. However, this distribution spread which is larger than in our method, is compensated for, on average, by a better shape with a sharper distribution with EYE-EEG synchronization. For the cross-correlation, the lags between horizontal EOG and eye positions, are on average longer when synchronization is carried out using the EYE-EEG toolbox. It should be noted that even if the differences between the lags are significant for both methods, they remain small with EYE-EEG synchronization (around one sample on average). However, it is still essential to have the lowest possible number of errors and a small lag in order to limit any potential bias when interpreting the results, especially in experiments using methods aimed at obtaining high-quality temporal resolution. Based on these qualitative criteria, the two synchronization methods are of similar overall quality, ideally when no pauses occur. The EYE-EEG method relies on only one regression, whereas many regressions can be implemented in our method (i.e., the number of regressions is equal to the

number of recordings separated by pauses). This limits the use of the EYE-EEG method to experiments without pauses.

Finally in keeping with current trends, and with the demands of science, the proposed synchronization algorithm was implemented in C++ language. Consequently, it has a very rapid execution time (about 15 s for one hour of acquisition). The resulting software in the form of a DOS application has been extensively and successfully tested in a number of situations. The software provided is easy to handle -a tutorial with instructions for users is provided in the “Supplementary Material”- and it does not require any graphical environment. Moreover, it can be easily extended by developing in front-end, data conversion functions which enables it to deal with the increasingly large number of manufacturers.

The development of robust procedures of synchronization is very important, because joint analysis of multimodal datasets aims to combine the complementary aspects of each modality in such a way that there is an added benefit compared to analyzing and interpreting each dataset separately. In addition, and in view of the growing number of EFRP experiments, this could create substantial added value for neuroscience applications. Multimodal data integration could lead to a more comprehensive view of brain processes and structures (Uludağ & Roebroeck, 2014), and the next step could be to develop an online solution with the same qualities of flexibility and robustness in the near future.

794

795 **Acknowledgments**

796 With a great deal of emotion, we dedicate this article to Gelu Ionescu, research engineer at the
797 GIPSA-Lab, who was at the origin and the development of the vast majority of this work, and who died in
798 a climbing accident a few days after his retirement. May this manuscript be a reflection of our eternal
799 gratitude for his investment in his daily work and his kindness.

800

801 **Funding**

802 This work was funded in part by a grant from the French National Research Agency (ANR) under
803 the project GAZE&EEG (ANR-09-BLAN-0330) and a grant from the LabEx PERSYVAL-Lab (ANR-11-
804 LABX-0025-01)

805 **Open Practices Statement**

806 The executable program and code as well as datasets (coming from eye-tracker and EEG acquisition
807 devices, according to the examples explained in this article) are available in the Zenodo repository:
808 http://doi.org/10.5281/zenodo.4897128

809

810 **References**

811 Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search
812 tool. *Journal of Molecular Biology*, *215*(3), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2
813 Bergroth, L., Hakonen, H., & Raita, T. (2000). A survey of longest common subsequence algorithms.
814 *Proceedings Seventh International Symposium on String Processing and Information Retrieval*, 39–

48.

Delorme, A., & Makeig, S. (2004). EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, *134*(1), 9–21. https://doi.org/10.1016/j.jneumeth.2003.10.009

Deriche, R. (1990). Fast Algorithms for Low-Level Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *12*(1), 78–87.

Devillez, H., Guyader, N., & Guérin-Dugué, A. (2015). An eye fixation-related potentials analysis of the P300 potential for fixations onto a target object when exploring natural scenes. *Journal of Vision*, *15*(13), 1–31. https://doi.org/10.1167/15.13.20

Dimigen, O., Sommer, W., Hohlfeld, A., Jacobs, A. M., & Kliegl, R. (2011). Coregistration of eye movements and EEG in natural reading: Analyses and review. *Journal of Experimental Psychology: General*, *140*(4), 552–572. https://doi.org/10.1037/a0023885

Ehinger, B. V., & Dimigen, O. (2019). Unfold: An integrated toolbox for overlap correction, non-linear modeling, and regression-based EEG analysis. *PeerJ*, *2019*(10), 0–33. https://doi.org/10.7717/peerj.7838

Engbert, R., & Kliegl, R. (2003). Microsaccades uncover the orientation of covert attention. *Vision Research*, *43*(9), 1035–1045. https://doi.org/10.1016/S0042-6989(03)00084-1

Frey, A., Ionescu, G., Lemaire, B., López-Orozco, F., Baccino, T., & Guérin-Dugué, A. (2013). Decision-making in information seeking on texts: An Eye-Fixation-Related Potentials investigation. *Frontiers in Systems Neuroscience*, *JUL*. https://doi.org/10.3389/fnsys.2013.00039

Frey, A., Lemaire, B., Vercueil, L., & Guérin-Dugué, A. (2018). An Eye Fixation-Related Potential Study in Two Reading Tasks: Reading to Memorize and Reading to Make a Decision. *Brain Topography*, *31*(4), 640–660. https://doi.org/10.1007/s10548-018-0629-8

838 Guérin-Dugué, A., Roy, R. N., Kristensen, E., Rivet, B., Vercueil, L., & Tcherkassof, A. (2018). Temporal
839 dynamics of natural static emotional facial expressions decoding: A study using event- and eye
840 fixation-related potentials. *Frontiers in Psychology*, *9*(JUN), 1–19.
841 https://doi.org/10.3389/fpsyg.2018.01190
842 Hoogeboom, P. J. (2003). Off-line synchronization of measurements based on a common pseudorandom
843 binary signal. *Behavior Research Methods, Instruments, & Computers*, *35*(3), 384–390.
844 Jorge, J., Grouiller, F., Ipek, Ö., Stoermer, R., Michel, C. M., Figueiredo, P., van der Zwaag, W., &
845 Gruetter, R. (2015). Simultaneous EEG–fMRI at ultra-high field: artifact prevention and safety
846 assessment. *Neuroimage*, *105*, 132–144.
847 Jung, T. P., Makeig, S., Westerfield, M., Townsend, J., Courchesne, E., & Sejnowski, T. J. (2000).
848 Removal of eye activity artifacts from visual event-related potentials in normal and clinical subjects.
849 *Clinical Neurophysiology*, *111*(10), 1745–1758. https://doi.org/10.1016/S1388-2457(00)00386-2
850 Kamienkowski, J. E., Ison, M. J., Quiroga, R. Q., & Sigman, M. (2012). Fixation-related potentials in
851 visual search: A combined EEG and eye tracking study. *Journal of Vision*, *12*(7), 1–20.
852 https://doi.org/10.1167/12.7.4
853 Körner, C., Braunstein, V., Stangl, M., Schlögl, A., Neuper, C., & Ischebeck, A. (2014). Sequential effects
854 in continued visual search: Using fixation-related potentials to compare distractor processing before
855 and after target detection. *Psychophysiology*, *51*(4), 385–395. https://doi.org/10.1111/psyp.12062
856 Kristensen, E., Guerin-Dugué, A., & Rivet, B. (2017). Regularization and a general linear model for event-
857 related potential estimation. *Behavior Research Methods*, *49*(6), 2255–2274.
858 https://doi.org/10.3758/s13428-017-0856-z
859 Kristensen, E., Rivet, B., & Guerin-Dugué, A. (2017). Estimation of overlapped Eye Fixation Related
860 Potentials: The General Linear Model, a more flexible framework than the ADJAR algorithm. *Journal*

of Eye Movement Research, *10*(1), 1–27. https://doi.org/10.16910/jemr.10.1.7

Liu, Z., Ding, L., & He, B. (2006). Integration of EEG/MEG with MRI and fMRI in Functional Neuroimaging. *IEEE Engineering in Medicine and Biology Magazine: The Quarterly Magazine of the Engineering in Medicine & Biology Society*, *25*(4), 46–53.

Nikolaev, A. R., Meghanathan, R. N., & van Leeuwen, C. (2016). Combining EEG and eye movement recording in free viewing: Pitfalls and possibilities. *Brain and Cognition*, *107*, 55–83. https://doi.org/10.1016/j.bandc.2016.06.004

Nikolaev, A. R., Pannasch, S., Ito, J., & Belopolsky, A. V. (2014). Eye movement-related brain activity during perceptual and cognitive processing. *Frontiers in Systems Neuroscience*, *8*(1 APR), 2013–2014. https://doi.org/10.3389/fnsys.2014.00062

Pak, W. (2017). Ultra-low-power media access control protocol based on clock drift characteristics in wireless sensor networks. *International Journal of Distributed Sensor Networks*, *13*(7). https://doi.org/10.1177/1550147717722155

Picton, T. W., Bentin, S., Berg, P., Donchin, E., Hillyard, S. A., Johnson, R., Miller, G. A., Ritter, W., Ruchkin, D. S., Rugg, M. D., & Taylor, M. J. (2000). Guidelines for using human event-related potentials to study cognition: Recording standards and publication criteria. *Psychophysiology*, *37*(2), 127–152. https://doi.org/10.1017/S0048577200000305

Rosenkranz, K., & Lemieux, L. (2010). Present and future of simultaneous EEG–fMRI. *Magnetic Resonance Materials in Physics, Biology and Medicine*, *23*(5), 309–316.

Shin, J., von Lühmann, A., Kim, D. W., Mehnert, J., Hwang, H. J., & Müller, K. R. (2018). Simultaneous acquisition of EEG and NIRS during cognitive tasks for an open access dataset. *Scientific Data*, *5*(180003), 1–16.

Uludağ, K., & Roebroeck, A. (2014). General overview on the merits of multimodal neuroimaging data

fusion. *NeuroImage*, *102*(1), 3–10.

Van Humbeeck, N., Meghanathan, R. N., Wagemans, J., Leeuwen, C., & Nikolaev, A. (2018). Presaccadic EEG activity predicts visual saliency in free-viewing contour integration. *Psychophysiology*, *55*, e13267. https://doi.org/10.1111/psyp.13267

Appendix

Synchronization of a dataset containing two pauses (scenario 1.3)

For a scenario 1.3 dataset containing two pauses, *Figure A1* shows the joint representation of the timestamps of each common trigger in both modalities before synchronization.

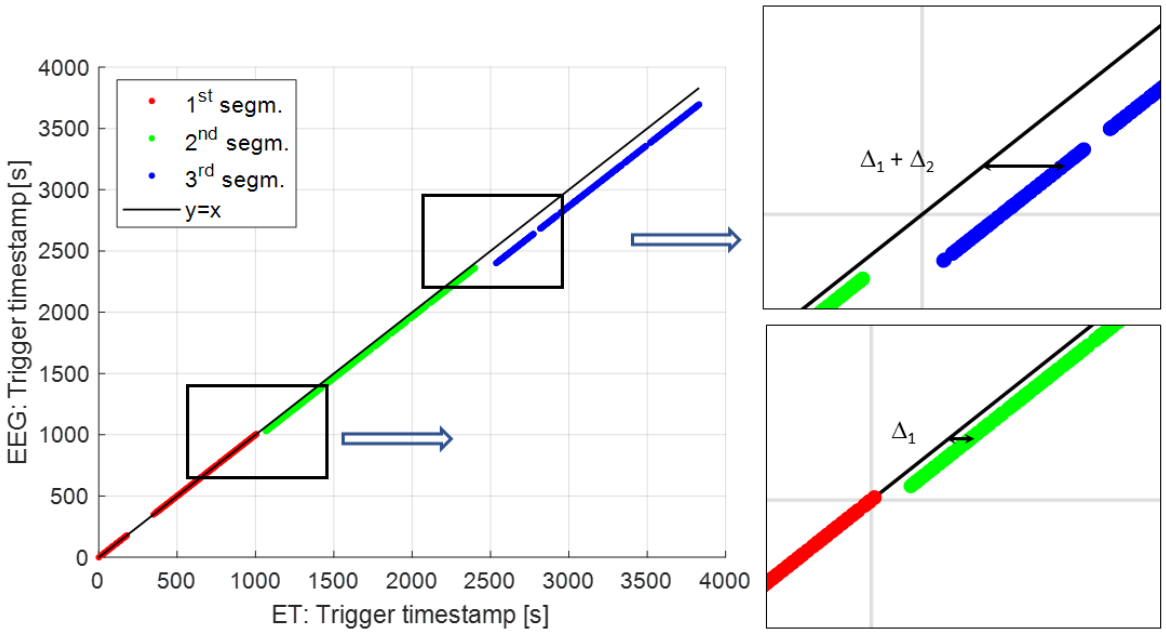

Figure A1. Joint representation of the timestamps of each common trigger in both modalities before synchronization. Δ1 and Δ2 are the observed time shifts that must be corrected by the drift and shift corrections.

The histogram of alignment errors after synchronization using the EYE-EEG method is shown in *Figure A2* A. Only two common triggers (the start and end triggers) were matched. For all other triggers,

the gap between their timestamps in each modality was larger than the tolerance parameter (by default, 4 samples). Only one regression was performed and the additional shifts resulting from the two pauses could not be compensated. This synchronization default was confirmed by the cross correlation between the horizontal eye position and the differential EEG signal computed from the F8 channel and the F7 channel (*Figure A2* B). The histogram of alignment errors, using the proposed method on the 1222 common triggers, is shown in *Figure 6* D. The additional shifts were found by performing the piecewise linear regression (shift correction). The cross correlation is shown in *Figure A2* C.

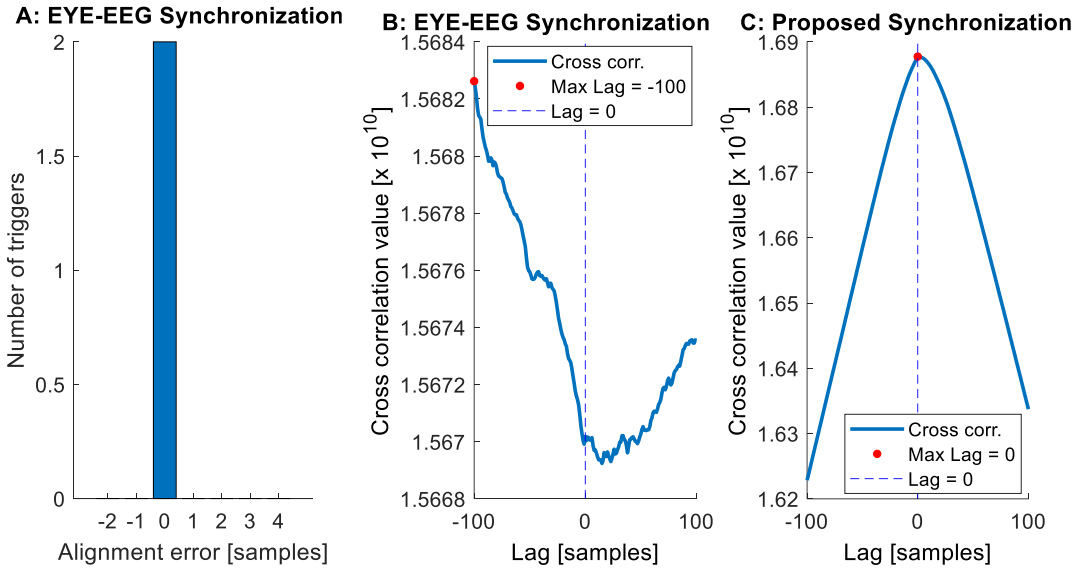

Figure A2. Synchronization of a scenario 1.3 dataset containing two pauses. (A) Histogram of alignment errors on the common triggers found by the EYE-EEG method. (B) Cross correlation between the horizontal eye position and the F8-F7 EEG signals, for the EYE-EEG synchronization and (C) for the proposed synchronization