A brain computer interface for robust wheelchair control application based on pseudorandom code modulated Visual Evoked Potential
Mohebbi, Ali; Engelsholm, Signe K.D.; Puthusserypady, Sadasivan; Kjaer, Troels W.; Thomsen, Carsten Eckhart; Sorensen, Helge B.D.

Published in:
37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC

DOI:
10.1109/EMBC.2015.7318434

Publication date:
2015

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
A Brain Computer Interface for Robust Wheelchair Control Application Based on Pseudorandom Code Modulated Visual Evoked Potential

Ali Mohebbi\textsuperscript{1}, Signe K.D. Engelsholm\textsuperscript{1}, Sadasivan Puthusserypady\textsuperscript{1}, Troels W. Kjaer\textsuperscript{2}, Carsten E. Thomsen\textsuperscript{3}, Helge B.D. Sorensen\textsuperscript{1}

Abstract—In this pilot study, a novel and minimalistic Brain Computer Interface (BCI) based wheelchair control application was developed. The system was based on pseudorandom code modulated Visual Evoked Potentials (c-VEPs). The visual stimuli in the scheme were generated based on the Gold code, and the VEPs were recognized and classified using subject-specific algorithms. The system provided the ability of controlling a wheelchair model (LEGO\textsuperscript{®} MINDSTORM\textsuperscript{®} EV3 robot) in 4 different directions based on the elicited c-VEPs. Ten healthy subjects were evaluated in testing the system where an average accuracy of 97\% was achieved. The promising results illustrate the potential of this approach when considering a real wheelchair application.

I. INTRODUCTION

Brain Computer Interface (BCI) is an interface between the brain and an external device (with the help of a computer), which enables signals from the brain to control the external device. It can be perceived as a communication scheme in which the user’s intention is converted to an output without involving the usual output pathways of peripheral nerves and muscles [1]. BCI research has opened up exhilarating options for disabled individuals, such as patients suffering from Amyotrophic Lateral Sclerosis (ALS), Acute Inflammatory Demyelinating Polyradiculoneuropathy (AIDP) and Spinal Cord Injury (SCI), to communicate with the outside world.

From the literature it can be seen that electroencephalogram (EEG) based BCIs have become more popular because of their non-invasive nature and easy usability. Several BCI schemes have been developed based on different types of EEGs, such as the Steady-State VEP (SSVEP), Event related potentials (ERPs), slow cortical potentials (SCPs) [2], [3], and code modulated VEPs (c-VEPs) [4], [5].

Among BCIs, c-VEP has been shown to be superior with respect to the accuracy, Information Transfer Rate (ITR) and the possibility of using many targets in applications [4], [6]. In a recent study, it has been specifically indicated that the c-VEP based BCI is the most suitable scheme for controlling a robotic device [7].

c-VEPs are repetitive potentials elicited in the occipital lobe of the brain, when the person is focusing on a visual stimulus, flickering in a repetitive pattern. Different targets flicker with the same pattern but shifted in time. By averaging a number of c-VEPs, and comparing them with predefined templates, it is possible to identify the target which the subject is focusing on [4].

The main goal of this pilot study was to investigate the option for a paralyzed person to control his/her own wheelchair by the use of a c-VEP based BCI. A system was developed with the application of controlling a LEGO\textsuperscript{®} MINDSTORM\textsuperscript{®} EV3 robot (henceforth referred to as the robot), representing a model of a wheelchair to move in 4 directions; left, right, forward and backward. On a monitor, 4 flickering targets presented the possible steering options. In a real application it would be expected that user’s focus is not constantly directed at the stimuli. Hence, it was desirable to minimize the number of targets selected when in fact no targets were focused on. In this case, a subject-specific classification algorithm was developed minimizing false positive classifications of the directions.

In a number of recent studies, BCI based wheelchair control has been investigated. However, they are based on either the SSVEP or ERP [8], [9]. In our work, a c-VEP based BCI scheme using Gold code as stimulus sequence was developed which, to the best of our knowledge, is the first of its kind for such an application.

II. MATERIALS AND METHODS

A. System configuration

A schematic of the developed c-VEP based BCI for wheelchair control is presented in Fig. 1. It includes visual stimuli presented on a 60 Hz LED monitor, EEG acquisition, signal processing with feature extraction, classification algorithm, visual feedback (i.e. target marked red) and wireless communication with the robot.

EEG signals were recorded from the subjects using the g.USBamp amplifier from g.tech (Guger Technologies), which is compatible with MATLAB (The MathWorks, Inc.). The amplifier was connected to the computer via a USB port. Three Ag/AgCl electrodes were placed on the scalp at Oz (active channel), Pz (reference) and Fpz (ground) locations in accordance with the international 10-20 system [10]. The electrode impedances were kept low (<5 k\Ω). The recorded EEG signals were band-pass filtered (5-30 Hz) using an 8th order Butterworth filter and were sampled at 600 Hz.

B. Visual stimuli

The visual stimuli were presented using the Cogent Graphics Toolbox in MATLAB\textsuperscript{1}, which has the ability to synchro-

\textsuperscript{1}Technical University of Denmark, Department of Electrical Engineering, Østeds Plads, 2800 Kgs. Lyngby, Denmark. Corresponding author: ali.m92@hotmail.com

\textsuperscript{2}Roskilde University Hospital, Department of neurology, Køgevej 7-13, 4000 Roskilde, Denmark

\textsuperscript{3}Department of Odontology, University of Copenhagen, Nørre Allé 20, 2200 Copenhagen, Denmark

\textsuperscript{1}Downloaded from http://www.vislab.ucl.ac.uk/cogent.php
nize the stimuli with the monitor frame rate. The stimuli were generated from a pseudo-random binary code, (0,1) corresponding to 2 inverse images of a chess-board pattern with an arrow in the middle (see Fig. 1). Hence, every time the screen refreshed itself, the next number (1 or 0) was read from the code, and the corresponding image was presented on the monitor. When the end of the code was reached, one stimulus cycle was completed, and the algorithm repeated itself starting from the first bit in the code. Since only 4 targets were needed in our application, a code of 15 bits was considered appropriate, with the targets shifted by 3 bits in the code with respect to each other. A set of 15 bits Gold codes were generated by performing the modulo 2 addition between two separate 15 bits Maximum Length Sequences (MLS) in their various phases (i.e. translated into all relative positions) [11]. After preliminary experiments including both MLS and Gold codes, a specific Gold code was chosen as the preferred stimulus sequence on the basis of performance.

C. Training session

When using c-VEP, a training session to generate individual templates is necessary. Here, the subjects were asked to look at the forward target flickering for 50 cycles (around 13 seconds), while the other targets were blinded. A template, $T_0(n)$, $n = 1, 2, ..., N$, for this target was generated by averaging the c-VEPs elicited in the EEG, hence providing one single c-VEP with high signal to noise ratio. Since the flickering targets elicit the same c-VEP but shifted in time (samples), templates for the remaining targets could be obtained by shifting $T_0(n)$ circularly as [6],

$$T_k(n) = T_0\left(n + \frac{f_s}{f_r} \cdot k\right), \quad k = 0, 3, 6, 9,$$

where $f_r$ is the frame rate, $f_s$ is the sampling frequency and $k$ is the number of bits with which the target is shifted in the 15 bits Gold code, relative to the forward target. Figure 2 illustrates the shifting procedure to generate $T_3(n)$ from $T_0(n)$. After the templates were generated, a short test was conducted to determine the optimal classifier thresholds for the individual subjects.

D. Timestamps

In order to detect and average c-VEPs in the EEG, two time-stamps were collected at the end of each completed stimulus cycle; one from the CPU and the other from the sampled EEG. It was discovered that the time-stamps collected from the CPU was consistent within runs, but were shifted between runs. With the EEG derived time-stamps, it was the opposite. It was indeed necessary to have equal time offsets when comparing templates with real-time data. This was ensured by subtracting the average offset (mean of the disparity between the two types of time-stamps) from the time-stamps collected from the CPU.

E. Classification algorithm

A subject-specific classification algorithm was developed to detect which target the subjects were focusing on. As depicted in Fig. 3, it was based on the correlation values, as determined by the pearson product-moment correlation coefficient [12]:

$$C_k = \sum_{n=1}^{N} \frac{\hat{T}_k(n) \cdot \hat{A}(n)}{\left[\sum_{n=1}^{N} \hat{T}_k(n) \cdot \sum_{n=1}^{N} \hat{A}(n)\right]^{1/2}},$$

where $\hat{T}_k(n) = T_k(n) - \overline{T}_k$ and $\hat{A}(n) = A(n) - \overline{A}$. Here, $A(n)$ is the data to be classified, $\overline{A}$ is the mean of $A(n)$, and $\overline{T}_k$ is the mean of the $k$’th template. This resulted in 4 correlation values, one for each target, with values between 1 and -1. The limits (1 and -1) represent the two sequences being either identical or reverse, respectively.

For each 8 stimuli cycles (2s or one run of EEG recording) presented on the monitor, the recorded c-VEPs were averaged, correlated and fed to the classifier. Here, the primary and secondary conditions were applied in the classification algorithm as illustrated in Fig. 3. Classification based on the primary condition represents an obtained correlation value higher than the subject-dependent primary threshold (determined in the training session). If this condition was met, the target was identified immediately, and the corresponding command was sent to the robot. If not, the secondary conditions were taken into account, based on the summation of the two most recent sets of correlations. In this case, two sub-conditions were applied. The first condition, similar to the primary condition was, that the highest summed correlation

![Fig. 1. Schematic representation of the BCI system configuration](image1)

![Fig. 2. The template of the backward target ($T_3$) can be obtained by shifting the template of forward target ($T_0$) circularly.](image2)
Fig. 3. Schematic representation of classifier algorithm with the primary and secondary conditions. Two seconds of EEG recording corresponds to 8 stimuli cycles displayed on the monitor. The plots show the templates (red curves) and averaged data to be classified (blue curves). Correlations for each target are shown in the top. In this case, the correlation for the backward target meets the primary condition. F: forward, B: backward, L: left, R: right, C: correlation coefficient.

had to exceed a subject-dependent secondary threshold. In addition, the difference between the highest and the second highest summed correlation had to exceed another preset threshold. If neither of these conditions were met, the most recent set of correlations were appended to memory and another 8 stimuli cycles were presented again.

The purpose of combining primary and secondary conditions was to decrease the false positives and increase the accuracy requiring an eventual extra run of stimulation and EEG recording.

F. Experiments

Evaluation of the system was based on the training session with construction of templates, followed by offline and online sessions. In the offline sessions, the robot was not attached. The subjects were asked to focus 4 times on each target in a random manner, until the intended target (or a wrong one) was recognized by the system. The procedure was followed by one minute with the stimuli turned off to determine the number of completely random identifications. This session was repeated twice, followed by two online sessions in which the subjects steered the robot through a figure 8 shaped track around two poles in a field, starting from a red square as seen in Fig. 4. When a target was identified, the stimuli paused while the robot moved 20 cm in the desired direction or turned to the requested side by 45 degrees. Depending on which of the 4 targets was identified by the system, the corresponding target/square turned red, providing visual feedback as shown in Fig. 1. This was followed by a short pause of 1.5 s before the stimuli proceeded flickering, giving the subjects a brief moment to decide the next move. During the sessions, subjects were asked to inform whether the system made a false identification.

With the same steering settings, the subjects controlled the robot through the same path using keyboard arrows. A pause of 1.5 s between commands still applied, to ensure similar conditions. The experimental procedures on the subjects were approved by the Department of Electrical Engineering, Technical University of Denmark (DTU).

III. RESULTS AND DISCUSSION

Twelve healthy subjects (11 males and 1 female, with an age range of 20-25), with normal or corrected to normal vision participated in the study. The subjects were seated in front of the monitor (60 cm distance), simultaneously being able to observe the robot on the track as shown in Fig. 4. The average results of each subject in the offline and online experiments are listed in Table I and II, respectively. Overall mean (μ) and standard deviation (σ) for each parameter were calculated based on these values. Two of the subjects (highlighted with gray in the tables) deviated by more than two standard deviations from the overall mean value in the offline sessions with respect to the Time Per Identification (TPI) (subject 11) and the accuracy (subject 12) - the most

<table>
<thead>
<tr>
<th>Subject</th>
<th>Accuracy (%)</th>
<th>TPI (s)</th>
<th>Rand. targets in one minute</th>
<th>Time per rand. target (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>96.88</td>
<td>2.58</td>
<td>4.50</td>
<td>9.78</td>
</tr>
<tr>
<td>2</td>
<td>93.75</td>
<td>2.27</td>
<td>2.50</td>
<td>16.46</td>
</tr>
<tr>
<td>3</td>
<td>100.00</td>
<td>2.46</td>
<td>5.00</td>
<td>10.08</td>
</tr>
<tr>
<td>4</td>
<td>93.75</td>
<td>2.39</td>
<td>3.00</td>
<td>18.65</td>
</tr>
<tr>
<td>5</td>
<td>100.00</td>
<td>2.27</td>
<td>3.50</td>
<td>13.61</td>
</tr>
<tr>
<td>6</td>
<td>96.88</td>
<td>2.14</td>
<td>4.00</td>
<td>12.10</td>
</tr>
<tr>
<td>7</td>
<td>100.00</td>
<td>2.52</td>
<td>4.00</td>
<td>12.60</td>
</tr>
<tr>
<td>8</td>
<td>96.88</td>
<td>2.46</td>
<td>5.00</td>
<td>8.73</td>
</tr>
<tr>
<td>9</td>
<td>100.00</td>
<td>2.46</td>
<td>6.00</td>
<td>8.23</td>
</tr>
<tr>
<td>10</td>
<td>100.00</td>
<td>2.27</td>
<td>3.50</td>
<td>14.03</td>
</tr>
<tr>
<td>11</td>
<td>90.63</td>
<td>3.15</td>
<td>3.50</td>
<td>13.95</td>
</tr>
<tr>
<td>12</td>
<td>75.00</td>
<td>2.39</td>
<td>4.50</td>
<td>10.71</td>
</tr>
</tbody>
</table>

\[
\mu \pm \sigma = 95.31 \pm 7.11, 2.45 \pm 0.26, 4.08 \pm 0.97, 12.41 \pm 3.14
\]

Corrected \[
\mu \pm \sigma = 97.81 \pm 2.57, 2.38 \pm 0.14, 4.10 \pm 1.05, 12.43 \pm 3.38
\]
TABLE II
AVERAGE RESULTS OF ONLINE SESSIONS

<table>
<thead>
<tr>
<th>Subject</th>
<th>Accuracy (%)</th>
<th>TPI (s)</th>
<th>Total time w. BCI (s)</th>
<th>Total time w. keyboard (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>87.22</td>
<td>2.82</td>
<td>167</td>
<td>71</td>
</tr>
<tr>
<td>2</td>
<td>100.00</td>
<td>2.10</td>
<td>107</td>
<td>68</td>
</tr>
<tr>
<td>3</td>
<td>98.28</td>
<td>2.08</td>
<td>118</td>
<td>61</td>
</tr>
<tr>
<td>4</td>
<td>93.23</td>
<td>2.48</td>
<td>138</td>
<td>61</td>
</tr>
<tr>
<td>5</td>
<td>98.21</td>
<td>2.52</td>
<td>138</td>
<td>70</td>
</tr>
<tr>
<td>6</td>
<td>100.00</td>
<td>2.21</td>
<td>148</td>
<td>62</td>
</tr>
<tr>
<td>7</td>
<td>98.15</td>
<td>3.05</td>
<td>142</td>
<td>62</td>
</tr>
<tr>
<td>8</td>
<td>100.00</td>
<td>2.98</td>
<td>135</td>
<td>66</td>
</tr>
<tr>
<td>9</td>
<td>96.30</td>
<td>2.28</td>
<td>123</td>
<td>71</td>
</tr>
<tr>
<td>10</td>
<td>100.00</td>
<td>2.71</td>
<td>135</td>
<td>63</td>
</tr>
<tr>
<td>11</td>
<td>90.21</td>
<td>4.26</td>
<td>209</td>
<td>67</td>
</tr>
<tr>
<td>12</td>
<td>79.61</td>
<td>2.35</td>
<td>212</td>
<td>62</td>
</tr>
</tbody>
</table>

\[ \mu \pm \sigma \]

Corrected Accuracy: 97.14±4.09

TPI: 2.52±0.36

Total time w. BCI: 134.8±16.3

Total time w. keyboard: 65.5±4.2

IV. Conclusion

As the main objective of this pilot study, we managed to develop a functional c-VEP based BCI system, demonstrating the possibility to steer a simulated wheelchair in real-time by the use of visual acuity. With the implementation of a simple, subject-specific classifier, an average accuracy of 97% was achieved when controlling a LEGO® MINDSTORM® EV3 robot, representing a wheelchair, in 4 different directions. 10 out of 12 subjects were able to control the robot satisfactorily through a figure 8 shaped track yielding an encouraging performance when considering its implementation into a real wheelchair application. In addition, this study indicates the usefulness of the Gold code as an alternative to the MLS, which is currently the most widely used pseudo-random sequence in c-VEP based BCIs [4]. The promising results accomplished in this pilot study aims towards further study and development of the system.

ACKNOWLEDGMENT

The authors would like to thank Adnan Vilic for his support throughout this study. We also appreciate Jaman Ahmad and Jacob Dahl Dybbro for their effort and assistance in addition with the subjects who volunteered to participate.

REFERENCES


important parameters considered. They were excluded from the analysis to provide consistency.

In the offline sessions (Table I), the remaining subjects achieved an average accuracy of 97.81±2.57%, and an average TPI of 2.38±0.14 s (i.e. seconds that pass from the time the stimuli starts flickering until the target is identified). The average time that passed before random target identifications occurred is generally high compared to the average TPI with a ratio of 5.2 between the means. However, the fact that 4.1±1.05 targets were randomly identified each minute indicates the necessity of improving the system by e.g. expanding with a stop button among other safety improvements in an actual wheelchair application.

In the online sessions (Table II), the subjects achieved an average accuracy of 97.14±4.09%, and an average TPI of 2.52±0.36 s. A decrease in accuracy and increase in TPI was expected, since the subjects had to control the robot through a given track, avoid obstacles, focus on the stimuli and anticipate the next direction in 1.5 s. TPI was increased in the online session for 7 out of the 10 subjects. However, a one-sided paired t-test with 5% significance level revealed no difference between performance in the offline and the online sessions, with respect to accuracy and TPI, which indicates the stability of the BCI system.

The average time to finish the track with the keyboard was 65.5 s, while it took 134.8 s or about twice as long, when using the BCI system. The extra time required is also caused by the challenge of focusing on multiple elements, and for this reason the system performance is considered satisfactorily.

With some subjects however, the system made false classifications on specific targets, e.g. when the subject wanted to move forward, the target for right movement was identified instead. With these subjects, the template, and hence the c-VEP, showed some periodicity, which explains this occurrence.