**Multimedia Tools and Applications manuscript No.**
(will be inserted by the editor)

# Improving Object Segmentation by using EEG signals and Rapid Serial Visual Presentation

**Eva Mohedano · Graham Healy · Kevin McGuinness · Xavier Giró-i-Nieto · Noel E. O'Connor · Alan F. Smeaton**

Received: date / Accepted: date

**Abstract** This paper extends our previous work on the potential of EEG-based brain computer interfaces to segment salient objects in images. The proposed system analyzes the Event Related Potentials (ERP) generated by the rapid serial visual presentation of windows on the image. The detection of the P300 signal allows estimating a saliency map of the image, which is used to seed a semi-supervised object segmentation algorithm. Thanks to the new contributions presented in this work, the average Jaccard index was improved from 0.47 to 0.66 when processed in our publicly available dataset of images, object masks and captured EEG signals. This work also studies alternative architectures to the original one, the impact of object occupation in each image window, and a more robust evaluation based on statistical analysis and a weighted F-score.

**Keywords** Brain-computer interfaces · Electroencephalography · Rapid serial visual presentation · Object segmentation · Interactive segmentation · GrabCut algorithm

## 1 Introduction

The human brain is capable of processing audiovisual information in a way that clearly outperforms machines in most applications. The multimedia research community is constantly trying to simulate the brain's behaviour to leverage its innate computational ability. A deep understanding, however, of the human brain remains one of the greatest scientific challenges. Recent initiatives, such the Human Brain Project in Europe or the BRAIN Initiative in the United

E.Mohedano · G. Healy · K. McGuinness · N. E. O'Connor · A.F. Smeaton
Insight Center for data Analytics, Dublin City University, Ireland
E-mail: eva.mohedano@insight-center.org

X. Giró-i-Nieto
Image Processing Group, Universitat Politcnica de Catalunya, Spain

States, have identified its exploration as one of the grand challenges of our time.

Although humans consistently outperform computers in the semantic interpretation of multimedia signals [11], the computational and storage power of machines can be scaled and networked dramatically beyond individual human capacities. These two observations are the foundation of the human computational technologies, which exploit the best of both by defining collaborative strategies. The steady decrease in the cost of EEG (Electroencephalography) systems in recent years has made these non-invasive Brain-Computer Interfaces (BCIs) accessible beyond the traditional disciplines that typically availed of this technology [18,23]. Visual analysis is one such field, with recent publications exploring the potential of EEG signals for image retrieval [9,26,25] and object detection [3,13].

The use of brain-computer interfaces is, however, still limited, primarily because the motor (or speech) capabilities of most humans provide richer interaction methods than BCIs. For this reason, many current applications use BCIs as a secondary interaction source to complement another primary one, or as a tool for scientists to study human behaviour via EEG analysis [10]. Brain-computer interfaces, however, have the potential to be enormously beneficial for seriously impaired people, such as those affected by *Locked In Syndrome* (LIS). These individuals are paralysed of nearly all voluntary muscles, so are disabled from motion and speech. Vision is always intact, although in extreme cases even eye movement is restricted [1], in which cases BCIs represent the only opportunity to interact with the world.

Although a controversial discussion topic between neuroscientists, some authors claim to have observed consciousness with EEG devices on patients with persistent vegetative state [5], which may open a door to some level of interaction with them. For these reasons, and as explained in [7], *BCI systems hold great promise for effective basic communication capabilities through machines, e.g. by controlling a spelling program or operating a neuroprosthesis.* The use of EEGs for these type of assistive technologies has been previously explored in applications like letter-by-letter spelling [21] or the control of robots [2,19].

The objective of this work is to demonstrate that BCI interfaces are useful in tasks beyond spelling out words. We focus here on interaction with multimedia: specifically, object selection and segmentation in images. The capacity to perform such segmentation using a BCI interface potentially has both practical and creative applications, such as selection of specific objects for similarity search, and mixing objects from different sources to create a new composition. We propose a system capable of accurately selecting an object in an image in a manner that is completely hands-free, using only measured signals from an EEG interface. In this way, previous work exploring image retrieval (global image scale) [9,26,25] and object detection (coarse local scale) [3,13] are extended to a pixel-level object segmentation. This task is addressed by applying the human computation paradigm, using noisy EEG signals to seed the well-known GrabCut [22] segmentation algorithm.

This work extends our previous study [17] by modifying the EEG processing (i.e using a simple linear SVM kernel instead of RBF kernels for building the classification models and changing the way to downsample the feature vectors) to significantly improve classification models and the final segmentation. We also study the effect on the accuracy of our classification models when displaying different percentages of foreground pixels in the target windows and we also evaluate the quality of our probability maps (EEG maps) with a new measure proposed in [15] to evaluate foreground maps. Finally, the segmentation is performed with 4 different strategies: In the first we directly binarise the EEG maps and consider the mask as the final segmentation. The second consists of first filtering and then binarising the EEG maps and consider that as the segmentation, and the third and fourth consists of using the previous obtained maps (binarized and filtered) and used them to seed the segmentation algorithm Grabcut, so we can study the gain in combining our binary maps with Grabcut.

## 2 Related Work

Previous works combining BCI and computer vision [9, 25, 12] have been focused primarily on image retrieval and object detection. In such work images are presented to participants according to the *oddball paradigm*. This approach consists of presenting a “target” image among many “distractor” images via Rapid Serial Visual Presentation (RSVP) [24]. Although the presentation rate of the images is high, around 10Hz, a specific signature in the corresponding EEG signals is produced when the user observes the target images (or rare stimulus). This signature is known as a P300 wave and it is a kind of Event-Related Potential (ERP) associated to the process of recognising a relevant visual stimulus [14]. The waves primary characteristic is a positive peak in the EEG signal typically emerging around 300ms after a target visual stimulus is observed.

Two previous works describing a BCI system applied to image retrieval and detection were presented by Wang [25] and Healy [9]. In both cases the authors perform RSVP of images from known datasets at 10Hz to detect those images in which a specific object appears. The main difference between them is that in Wang’s paper the user is not asked to press any additional button when a target image is seen. Our work differs from these because it focuses on target windows (or regions) instead of target image detection. The most similar work to ours is Bigdely-Shamlo’s paper [3], in which satellite images are explored using local windows to detect those containing airplanes. Bigdely-Shamlo’s work, however, assumes that the object fits in a single window, while in our contribution objects are partially represented in an unknown number of windows.

In this study we do not use eye-tracking due to the high image presentation speeds employed not allowing time for useful eye movements. It is known, however, that P3-like responses do exist surrounding deployments of gaze (fixation) in images and that these can be used to detect local target stimuli which might not be apparent until after fixation has occurred [8].

## 3 System Architecture

We propose a system that aims to both detect and segment an object from an image using P3 brain responses that occur after observing a segment corresponding to a target-relevant image region. The idea is to transform the measured EEG responses into a map that gives an estimate of how probable it is that a particular region seen by the user contains the target object, and then to use this map to seed a segmentation algorithm. The construction of this map is based on EEG signal classification, as the electrical responses of the brain are known to differ when the user detects a target or rare stimulus in a RSVP scenario.

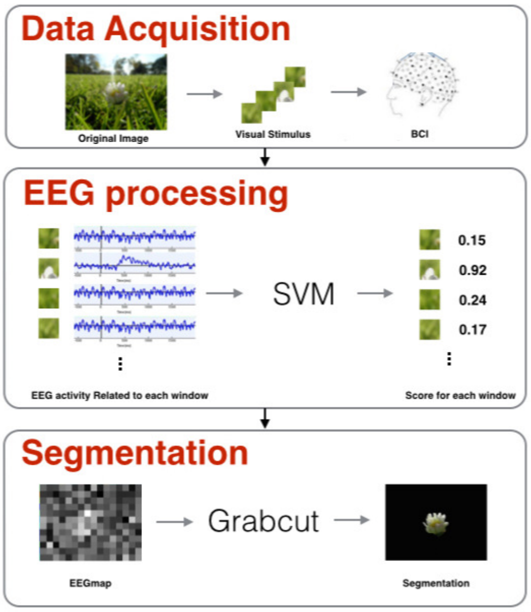

Fig. 1: System overview

Figure 1 illustrates the three primary stages of the proposed system:

1. **Data acquisition** (Section 4): in this stage we capture the brain signals related to the visual stimulus.
2. **EEG processing** (Section 5): pre-processing and classification are used to generate the probability maps for the object location. As these maps are built by using EEG analysis, they will be referred to as *EEG maps*.

3. **Segmentation** (Section 6): EEG maps are processed and 4 strategies to perform the final segmentation are evaluated: The first two consist in binarising and filtering the EEG maps, meanwhile the last two consist in combining the two different versions of the binary masks obtained to seed the GrabCut object segmentation algorithm [22].

   The following sections of the paper describe each stage in more detail.

## 4 Data acquisition

This section describes the experimental set-up used to capture the data. First, a new image dataset was created and each image partitioned in blocks of equal size. Each of these blocks are presented at a high rate. This stage was validated with a preliminary test with a single user, an important step before starting a larger campaign of data acquisition.

### 4.1 Image dataset

A novel dataset of 22 images was created to run the experimentation described in this paper. This dataset is publicly available at [1]. Given the exploratory nature of this work, the images were chosen to include a single object in a background of limited complexity. The dataset includes different configurations regarding the color, shape, and texture of the objects, as well as their relative similarity with the foreground, as shown in Figure 2. Each of the images has an associated ground truth for object segmentation in the form of a binary mask.

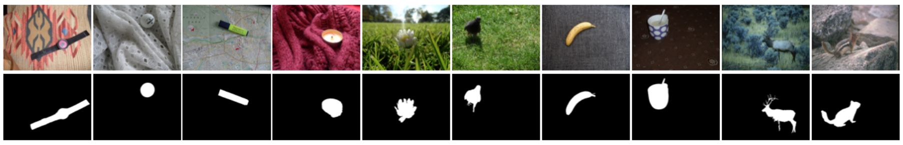

Fig. 2: Sample of 10 images of the dataset (first row) and their associated ground truth (second row). The last two images have been taken from *Berkeley Segmentation Dataset and Benchmark* (BSDB) [16].

### 4.2 Windows presentation

The goal of this stage is the generation of the visual stimuli in such a way that they generate different and measurable brain responses depending on whether

[1] `https://imatge.upc.edu/web/resources/eeg-signals-object-segmentation`

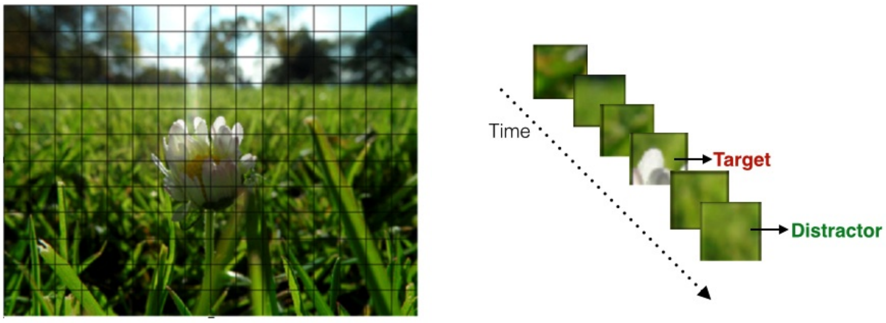

Fig. 3: Illustration of RSVP to randomly display different regions of an image.

they are related to target object or background pixels. The approach adopted is based on the Rapid Serial Visual Presentation (RSVP) [24] of the different windows that compose an image containing an object of interest. The approach follows the same idea described in the papers for image retrieval using BCI [9, 25, 12] but applied at local scale. This involves partitioning the images into 192 windows and displaying each of them in a fast and random succession (Figure 3). Given the homogeneous scale of the objects in the dataset and the amount of windows, these windows will usually only contain part of the object. In particular, the adopted ratio generated an average of 15% of windows containing parts of the object.

A 32-channel ActiChAmp EEG system with a sampling rate of 1kHz was used to capture EEG. 5 volunteers between 21 and 32 years old participated in the study. The electrodes were positioned according to the 10-20 system. The experiment was run in a quite electrically-shielded room. This room isolates both the participant and recording equipment in order to minimize the influence of electrical noise sources and other potentially distracting interferences on the user.

Image presentation in the experiments was carried out as followed. First, the entire image was displayed to the participant for five seconds. This allows the user to memorise the visual features of both object and background. Afterwards, the 192 windows of each image were presented at a rate of 5Hz. Each region is shown zoomed and centered on the screen. Preliminary experiments showed participants attention decreased with time. To minimise this effect, we asked participants to count the number of windows containing a part of the object.

### 4.3 Preliminary experiments

Acquiring EEG data on real users is both laborious and time consuming: in addition to the time required to actually perform the experiments (approximately one hour per user), it requires scheduling time with volunteers, equipment setup, and precise positioning of the various EEG sensors in a controlled environment. To ensure maximum benefit from each experiment trial, we decided to carry out a set of preliminary small-scale and simulated experiments. The objective of these experiments were: first, to establish whether classification of EEG signals with some reasonable degree of accuracy using our equipment and experiment setup is indeed feasible; second, to determine whether, given a imprecise classification of an EEG signal for a window, it is possible to use this to locate and segment the corresponding object from an image; and third, to guide us in making reasonable choices for the parameters such as the number and size of windows and their presentation rate. We include some details on these experiments here for reproducibility and to justify our design decisions. Positive results at this stage indicated that the system could indeed be effective and helped underpin the full-scale experiments.

### *4.3.1 Averaging of targets and distractors*

The first study focused on the temporal evolution of the EEG signal in those cases where this was captured for the presentation of a target or a distractor window. Given the noisy nature of EEG signals, the observation of any difference between two individual plots from the two classes is challenging. Nevertheless, this noise can be reduced by averaging several signals from the same class and, in this way, distinguish a clear Event Related Potential (ERP) waveform.

Figure 4 compares the same number of target (left) and distractor (right) signals captured in one electrode. The time span goes from one second before the visual stimulus to two seconds after it. The behaviour on the target reactions is different to the distractors, evidenced by a peak around 500ms after the stimulus visualization, which is clearly noticed in the averaged waveform across all the target trials.

This first result provided the evidence that the adopted RSVP strategy was capable of generating different and measurable brain responses for the two classes of windows. It must be made clear, however, that the future sections in the remainder of this paper do not apply any averaging strategy on the EEG signals associated to an image window. All results presented in later sections are based on the classification the EEG signal obtained with a single trial.

## 5 EEG Processing

This section describes the processing carried out on the EEG recordings to extract relevant information to identify the patches of the images in which part of the relevant object is located. The process contains two main steps:

1. Generating the feature vectors based on the EEG signals to represent each window.
2. Building a SVM model to predict a score to indicate how likely it is that a window contains part of a target object.

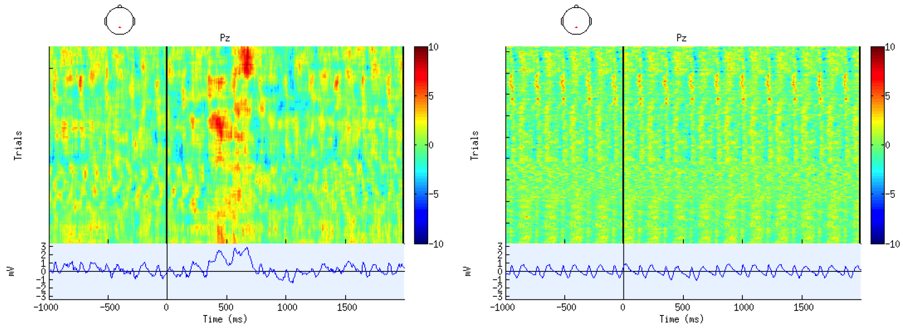

Fig. 4: One second before and two seconds after the visual stimulus recorded in the Pz channel for all participants (grand average). Shown are: the amplitudes of the brain waves (top), and the averaged values over all the waves for the target window (bottom-left), and distractor window epochs (bottom-right).

### 5.1 EEG feature vectors

First, the data was referenced to the average of all the 32 channels and the frequency rate was reduced from 1000Hz to 250Hz. After that, a band-pass filter from 0.1Hz to 20Hz was applied and the EEG activity related to one second before and two after each window presentation were selected (epochs). At this stage each image had 32 signals of 750 samples. Then, for each of the 32 signals, the period from 200ms to 1000ms was taken as the discriminant time region to discern between EEG responses of targets and distractors (see Figure 4). We selected the activity from 200ms to 1000ms after the stimulus presentation and we reduced the signal's sample rate from 250Hz to 20Hz, generating a 16 sample vector per channel. Each of the 16 samples per channel was the result of computing the average of 24 samples windows with 50% overlap between each other, which is a better strategy than linear interpolation of the signals as this approach has a tendency to be more adversely affected by high frequency components/noise in the EEG. It is known that lower frequencies (¡3Hz) of the EEG are primarily responsible for the generation of the P300 []. Finally, we build a single feature vector for the image as the concatenation of the 32 channels, generating a 512-dimension vector per window. The final feature vectors were normalized using l2 normalization.

### 5.2 Window classification

A model to predict the regions of interest within an image was generated for each user. We selected a linear kernel SVM with default parameters for that task due to the fact that no significant difference in performance was found when comparing the linear with the RBF SVM kernel used in [17] (t-test, $p = 0.814133$, sample size= 25) and the computation time was order of magnitudes shorter.

The EEG data for the 22 images was separated into 17 images for training and 5 for testing the model. Training and testing images were consistent across users. Thus, the training set consisted in an imbalanced set of 435 examples of targets and 2829 examples of distractors, respectively labeled with 1 and 0. The final model was tested on the separated 5 images, which contained a set of 130 targets and 830 distractors.

### 5.3 Evaluation of user performance

The performance of the models is evaluated in terms of Area Under the Curve of the Receiver Operating Characteristics (AUC-ROC). For each user, we cross validate the performance of their models by changing the 5 test and 17 train set of images 5 times (5-CV).

The final user performance is reported in Table 1 using the EEG procedure in our previous work (Old Pipeline) [17] but using a linear SVM instead of a RBF kernel. The procedure adopted in this current paper (New Pipeline) has shown a significant improvement in performance (One tailed t-test, $p = 0.006287$, sample size= 25) by increasing the mean ROC-AUC from 0.70 to a 0.79 with respect to the previous models.

Table 1: Mean ROC-AUC over the 5-CV of the models. Values are displayed with their associated stantard deviation.

| | User 1 | User 2 | User 3 | User 4 | User 5 | Mean ROC-AUC |
|---|---|---|---|---|---|---|
| Old Pipeline | .65±.02 | .73±.04 | .73±.05 | .71±.05 | .68±.02 | .70±.02 |
| New Pipeline | .74±.03 | .82±.01 | .85±.02 | .81±.05 | .73±.03 | .79±.02 |

### 5.4 EEG maps

The confidence scores provided by the classifier can be graphically represented as an image in the form of *EEG maps.*

This score represents the distance that separates the classified sample from the hyperplane [20]. Depending on the sign of this distance, the binary classifier assigns a target or distractor label. The maps are built by normalizing the values assigned to each window between 0 and 1 according to Equation 1:

$$X' = \frac{X - min(X)}{max(X) - min(X)}, \tag{1}$$

where $X'$ represents the normalized EEG map and $X$ the original EEG map.

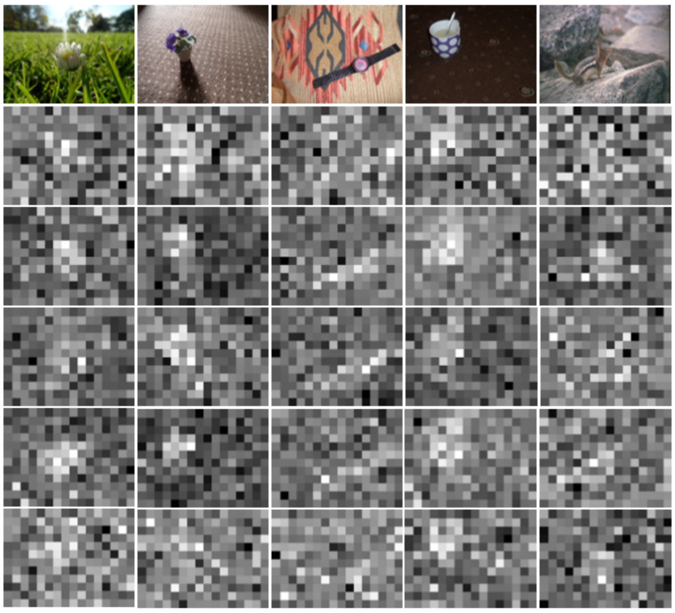

Fig. 5: EEG maps for a test set of images. The top row is the original images and the remaining rows are the generated EEG maps for five different participants, one in each row. Brighter pixels represent higher probabilities

The EEG maps obtained for the first set of testing images is displayed in Figure 5. As expected, the quality of the obtained EEG maps depends on the quality of the SVM models, so that the users with lower AUC also obtained the visually worst EEG maps (specially notable in the second column in the Figure 5).

To quantify the difference in quality between EEG maps, a measure proposed by Margolin et al. [15] was computed. The authors proposed an novel method to compare gray scale and binary maps with the ground truth mask. The proposed measure is a weighted version of the F-score ($F_w$) that, apart from considering the amount of true/false positives and negatives pixel labels in the foreground mask, also takes into account the relative position and relevance between pixels. For further information about this measure see [15].

### 5.5 Effects of the percentage of foreground in the target windows

The models presented in Section 5.4 have been trained considering all the windows containing one or more foreground pixels as targets. The approach is good enough to build a first noisy version of the EEG maps, but it does not take into account the variety on the kind of targets in terms of percentage of displayed foreground. For instance, Figure 6 presents an example of three target windows for a particular image. Although users have previously seen

Table 2: $F_w$ score for a test set if images (Figure5)

| | Image 1 | Image 2 | Image 3 | Image 4 | Image 22 | Avg±std |
|---|---|---|---|---|---|---|
| User 1 | .077 | .076 | .100 | .124 | .091 | .093±.020 |
| User 2 | .080 | .082 | .114 | .129 | .085 | .098±.022 |
| User 3 | .091 | .096 | .112 | .119 | .107 | .105±.011 |
| User 4 | .097 | .097 | .105 | .117 | .117 | .103±.009 |
| User 5 | .065 | .066 | .090 | .124 | .102 | .089±.025 |

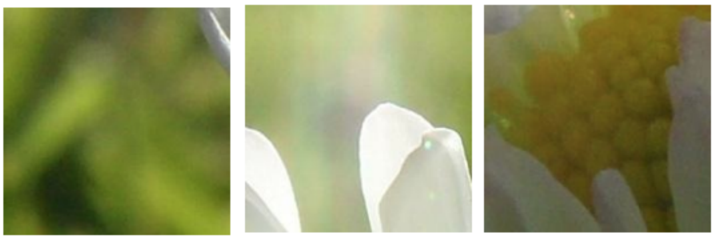

Fig. 6: Thee different target windows for an image that contains a flower as a target object. The window on the left contains around a 3% of foreground pixels, the one in the middle a 40% and the one in the right a 100%

the full image, it can be challenging to recognize a window containing just a small amount of foreground. In the same way, it can also be challenging to identify the object when the full window is based just on foreground pixels, without displaying any background. It seems reasonable to think that the windows containing partially object and background in similar proportion are the ones easier to detect, since they contain a patch of the object big enough to identify it without loosing the context information of the background.

To quantitatively address this issue, we ran an experiment training different models to detect different kind of targets. Specifically, 10 linear SVM were trained per user considering 10 different percentages of foreground ranges. (Model $n$, for $n$ in $[1, 10]$ considers foreground pixel percentages between $(10(n-1), 10n])$. The EEG signals related to the 22 images of the collection were used to train and test the models. The performance of the models was cross-validated 10 times, randomly selecting 30% and 70% for testing and training in each iteration.

Results in Figure 7 indicate that there is a difference in performance regarding the kind of targets. For all the users, either ranges with small foreground pixel percentages (less than 10%) or large percentage (more than 70%) perform slightly worse than inter middle ranges. Nevertheless, this effect can be influenced by the amount of available training samples per percentage range (Figure 9): When increasing the range of the percentage of foreground we are also decreasing the amount of available positive training examples. Notice though that when comparing the 0-10 and 10-20 bins, AUC increases despite using less training samples. This observation suggests that targets with less than 10% are indeed hard to identify.

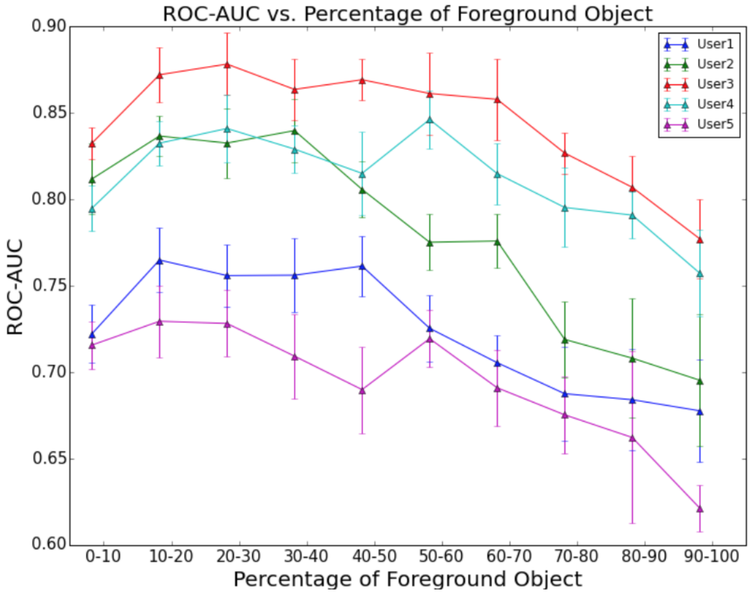

Fig. 7: Averaged ROC-AUC across the 10-CV iterations for each different percentage range. (The error bars are the standard deviation of each distribution.

## 6 Object segmentation

The EEG maps constructed in the previous section provide local information about how likely it is to find an object part in each window. The final segmentation requires a post-processing of the EEG maps to obtain a pixel-wise binary mask of the object location. Two approaches were tried for performing the final segmentation:

1. Segmentation by thresholding;
2. Segmentation with Grabcut.

The first approach consisted in setting a threshold to directly obtain a binary mask from the EEG maps and consider it the final segmentation. Additionally, this binarization was applied on a smoothed version of the EEG maps, aiming at reducing noise of the original maps. The second approach consisted in using the previously obtained binary masks as seeds for the well-known Grabcut segmentation algorithm [22].

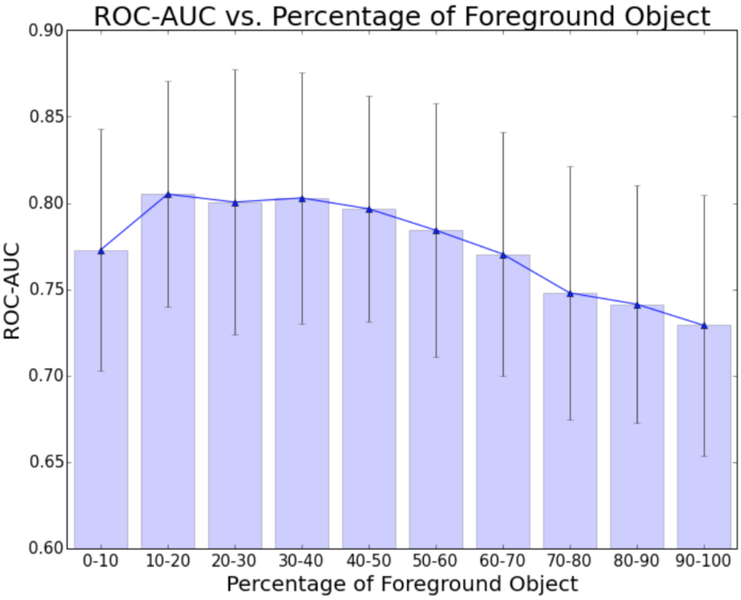

Fig. 8: Averaged ROC-AUC across the 10-CV iterations for each different percentage range. Results are averaged across all the users. The error bars are the standard deviation of each distribution.

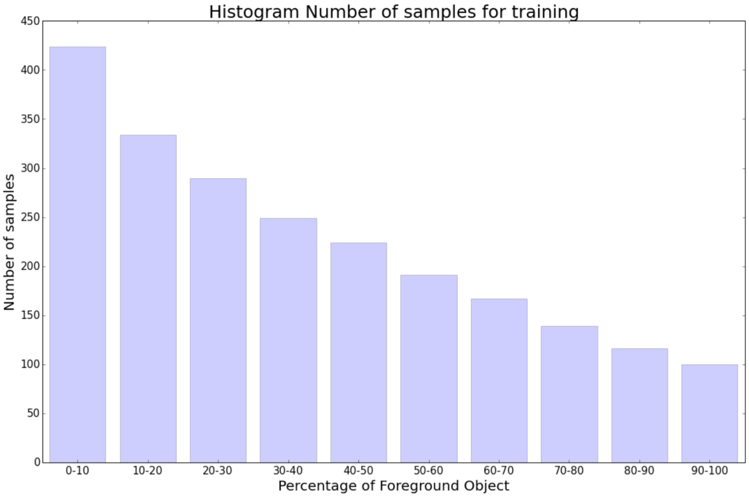

Fig. 9: Number of target training samples per percentage of foreground pixels range in the 17 set of training images.

For both procedures, EEG maps are generated after training the SVM model on 17 images. In this section results are reported considering always the same 17 and 5 images for training and testing. Nevertheless, in the final section we also cross-validate the performance of the system repeating the whole pipeline for different data splits into training and test. This allows us to generate the segmentation for all the images of the collection and also to validate the overall system performance. The different values for the segmentation parameters were learned on these training images based on the average performance of the 17 processed EEG maps.

The quality of the segmentation was evaluated with the Jaccard Similarity Index, a popular metric for object segmentation used, for example, in Pascal Visual Object Classes (VOC) Challenge [6]. This measure evaluates the similarity between the final segmentation and ground truth masks. The Jaccard Index has values between 0 an 1, with 1 the maximum similarity between the masks. The measure is defined as the intersection of the two final binary masks divided by the union of both masks:

$$J(A,B) = \frac{A \cap B}{A \cup B} \tag{2}$$

where $A$ is the segmentation mask and $B$ is the ground truth mask.

## 6.1 Segmentation by thresholding

In this subsection we describe the procedure of generating binary masks from the EEG maps obtained in Section 5.

### *6.1.1 Binarizing the EEG maps*

The simplest strategy to quantitatively assess the EEG maps in terms of object localization is to directly convert them into a binary mask. Such binarization is achieved by setting a threshold $\alpha$, which will consider as targets all those pixels in the EEG map which are higher than $\alpha$, and label as distractors all the rest. An optimal binarization threshold $\alpha_i$ was estimated for each individual user $i$ by averaging the $\alpha_{i,j}$ values that provided the highest Jaccard index for each training image $I_j$.

$$\alpha_{i,j} = \operatorname*{argmin}_{\alpha} J(M_{i,j}(\alpha), GT_j) \tag{3}$$

where $M_{i,j}$ is the EEG map thresholded by $\alpha$ for user $i$ and image $I_j$, and $GT_j$ is the ground truth mask for image $I_j$.

Visual results for this approach are presented in the left part of Figure 10. It is possible to see a high density set of windows labeled as target around the object location, especially for user 4.

Table 3 contains optimal thresholds to binarise the images of the test set, learned from the 17 EEG maps of training. It also contains the averaged Jaccard for the test set. In total, the approach provides a global Jaccard of 0.22, which points at the poor performance of a direct binarization on the EEG map.

Table 3: Final threshold per user obtained from the EEG maps for training and the final average value obtained applying the threshold on the test set. The Jaccard Index reported is the average of the 5 Jaccard indices of the test set, with their standard deviation.

| | User 1 | User 2 | User 3 | User 4 | User 5 | Mean User |
|---|---|---|---|---|---|---|
| $\alpha$ | .68 | .67 | .66 | .69 | .64 | .67±.02 |
| $J$ | .18±.09 | .22±.07 | .28±.07 | .28±.08 | .15±.08 | .22±.09 |

### *6.1.2 Filtering and binarization of EEG maps*

The binarization approach presented in the previous section presents a first limitation because of the block artefacts introduced by the window boundaries. The window contours do not need to match with the object ones, so in general this lack of resolution is partially responsible of the bad performance of the solution. In addition, the spatial relationship between the windows is completely ignored, without any contextual analysis that may provide coherence to the overall composition.

In this section, a low-pass filter is added before thresholding the maps to reduce block artefacts. With this filter, the isolated false positive windows of the background can be reduced and the high compact windows around the object will mutually reinforce. Equation (4) describes the filter mask (kernel) that is convoluted with the image. The $(x, y)$ values are the horizontal and vertical distances from the origin to a certain point of the kernel. The kernel takes standard deviation $\sigma$ as a parameter defining the spatial extension of the filter:

$$G(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \tag{4}$$

The Gaussian filtering and posterior binarization of the resulting EEG map requires defining the two parameters $\alpha$ and $\sigma$. As in the previous section, these were selected via minimizing the error over the training dataset. In this case, though, the Gaussian filtering changes the dynamic range of the EEG maps, which is no longer between 0 and 1. For this reason, the binarization threshold is not learned as an absolute value but as a normalised coefficient $p \in [0, 1]$ with respect to the dynamic range of the EEG map:

$$\alpha_{i,j}(p) = min(F_{i,j}) + p \cdot (max(F_{i,j}) - min(F_{i,j})), \tag{5}$$

where $F_{i,j}$ the filtered EEG map of user $i$ for image $I_j$.

The procedure used for optimisation was to select the parameters $(\sigma, \alpha)$ that generated the maximum averaged Jaccard Index over all the images of the training set. 70 values for sigma ($\sigma \in [0, 70]$) were tested to filter the EEG map. For each filtered map, 100 different values were tried by varying $p$ from 0 to 1, and the binarization threshold $\alpha_{i,j}$ that maximized the Jaccard was selected, as previously presented in Equation (3). Then, for each image an optimal combination $(\sigma, \alpha)$ that maximized the Jaccard was obtained. Finally, the parameters used in the test set were obtained by averaging the 17 pairs of optimal parameters computed for the training set.

The new binary masks shown in the right part of Figure 10 present in many cases a single patch located near the actual position of the object, with a shape which is much more natural than the sparse blocks generated in the left half of Figure 10. Quantitative results are presented in Table 4. We can see that, by filtering the maps, it is possible to nearly duplicate the quality of the final segmentation, obtaining an average Jaccard of 0.43. The reason is that it is possible to filter the noise introduced by the windows and, after binarising, a more accurate location and shape for the target object can be obtained.

Table 4: Averaged percentage (normalized to one) and $\sigma$ per user obtained from the train set and final Jaccard index obtained on the test set by using these parameters.

| | User 1 | User 2 | User 3 | User 4 | User 5 | Mean User |
|---|---|---|---|---|---|---|
| $\alpha$ | .76 | .76 | .76 | .72 | .79 | .75±.02 |
| $\sigma$ | 24.47 | 22.47 | 21.18 | 18.88 | 25.59 | 22.52±2.66 |
| $J$ | .35±.22 | .51±.23 | .51±.14 | .49±.11 | .29±.18 | .43±.19 |

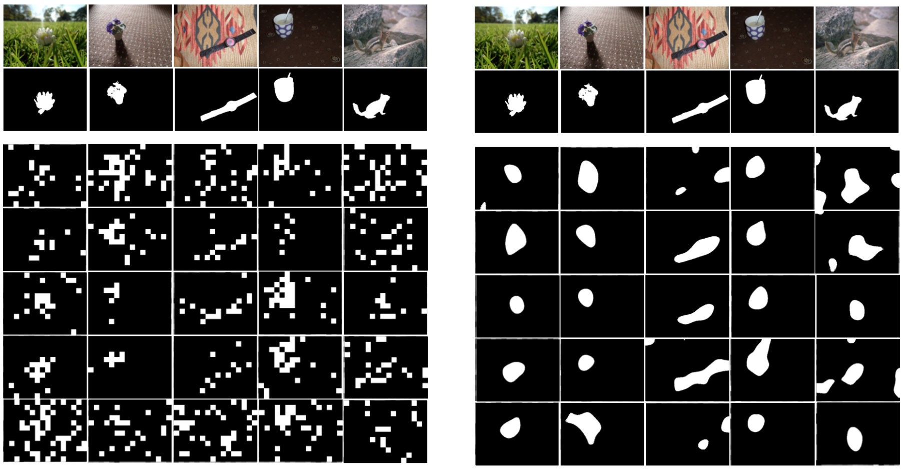

Fig. 10: Binary mask after filtering and thresholding the EEG maps. On the left, the the EEG maps thresholded, and on the right the EEG maps filtered and thresholded. Each row represent the final masks per user. First and second row are the original images and their ground truth masks.

### 6.2 Segmentation by Grabcut

The results obtained in the previous sections, based only on EEG data, already provide in many cases a rough estimation of the object location. The configuration explored in this section explores the synergy between BCI data and computer vision algorithms. The EEG maps (both the original and the filtered version), once they are binarised, can be used to seed an object segmentation algorithm that can exploit the spatial dependencies between neighbouring pixels. This way the computer vision algorithm is guided by the user in a noisy and approximate fashion.

The segmentation algorithm adopted in our work is GrabCut [22]. This technique performs a segmentation of an image based on a rough initial labeling defined by the user, typically by drawing a box around the target object. The pixels outside the box are initially considered as background and the pixels inside as unknown. GrabCut separately models the pixels labeled as background and the ones labeled as unknown by using a Gaussian Mixture Model (GMM). The unknown pixels are considered foreground pixels in the first iteration. Then, the two GMMs obtained are used to solve a minimization problem via min-cut and produce a first segmentation of the object. After the initial iteration, with the new labels for background and foreground, the GMMs are updated and the process is repeated until converge on the final segmentation. Our proposal here is to replace the drawn rectangle for the binarised EEG maps, where the white pixels and black pixels are the seed for foreground and background from where the algorithms start the optimization process.

Table 5 shows the final Jaccard when using the maps as a seed for Grabcut using the implementation included in OpenCV [4]. We found that by adding Grabcut to the pipeline it is possible to, on average, increase the accuracy on the final segmentation by 0.33 and 0.19 compared to the binarized and filtered and binarized maps, respectively. We note that even though the quality of the seed maps is different, the filtered version is significantly better than just binarizing the EEG maps (t-test, $p < 0.00001$, sample size= 125). When the maps are used with Grabcut, the final segmentation is similar, and also presents a high variance due to the fact that some images perform significantly better than others. We also include results when using Watershed algorithm instead of Grabcut with our best version of the binary EEG maps (Filtered version). We used the openCV implementation of the algorithm, setting the kernel for the erosion/dilation of the binary maps to 7x7 pixels and performing 3 iterations for each morphological operation to set the foreground and background markers. Unlike adding Grabcut to the pipeline, results show that when adding the Watershed algorithm the final segmentations did not present any significant gain, obtaining an equivalent Jaccard Index the one obtained by just using the Filtered maps.

Table 5: Jaccard Index when using the binarized and filtered and binarized EEG maps as a seed for Grabcut

| | User 1 | User 2 | User 3 | User 4 | User 5 | Mean User |
|---|---|---|---|---|---|---|
| *Binarized* | .18±.09 | .22±.07 | .28±.07 | .28±.08 | .15±.08 | .22±.09 |
| *Filtered* | .35±.22 | .51±.23 | .51±.14 | .49±.11 | .29±.18 | .43±.19 |
| Binarized+Grabcut | .51±.42 | .70±.40 | .79±.21 | .68±39 | .50±.44 | .63±.37 |
| Filtered+Grabcut | .54±.40 | .76±.21 | .69±.22 | .77±.22 | .57±.30 | .67±.27 |
| Filtered+Watershed | .38±.19 | .52±.18 | .56±.18 | .50±.18 | .35±.20 | .46±.18 |

## 7 Results

This section presents the results obtained when processing the presented the image dataset separately for each user, as well as combining their interaction to generate a higher quality segmentation.

As the number of images for testing the system is limited, a cross-validation is performed by switching the images in the test and training set 5 times and obtaining in this way the segmentation of all the dataset. That means that 5 different systems are generated following the described pipeline, where the 5 testing images are always independent from the training set.

### 7.1 Single user object segmentation

The averaged Jaccard Indexes per image are presented in Figure 11. In general, processed EEG maps obtain $0.20 \pm 0.10$ and $0.43 \pm 0.18$ for the thresholded and filtered and thresholded (*Binary* and *Filtered*) versions, respectively. When adding Grabcut, performance increases to $0.55 \pm 0.4$ and $0.66 \pm 0.32$. Although due to the high variability across images, this difference is not statistically significantly (t-test, $p = 0.27792$, sample size=125) which suggests that both processed versions perform similarly when combined with Grabcut, even though filtered versions are more accurate than the ones only thresholded.

Figure 12 presents the visual segmentation for five examples, as well as the intermediate stages. These results show that it is possible to successfully classify the brain response produced to detect different parts of a target object, and to produce useful information based on the EEG waves to locate the target object in the images.

Despite that Grabcut, in general, improves the segmentation obtained by the *Binary* and *Filtered* versions of the EEG maps, the algorithm not always improves results or, if there is an improvement, the final results are far from being the optimal ones. Figure 13 illustrates this effect for some images presented to user 3. Table 6 contains the Jaccard Index on the different steps of the processing. If we focus on the rows 2 and 3 of the Figure 13, one could think that, in the case of the *Binary* maps, the maps are too noisy to be used as a seed for Grabcut and that would be the reason of the low performance of

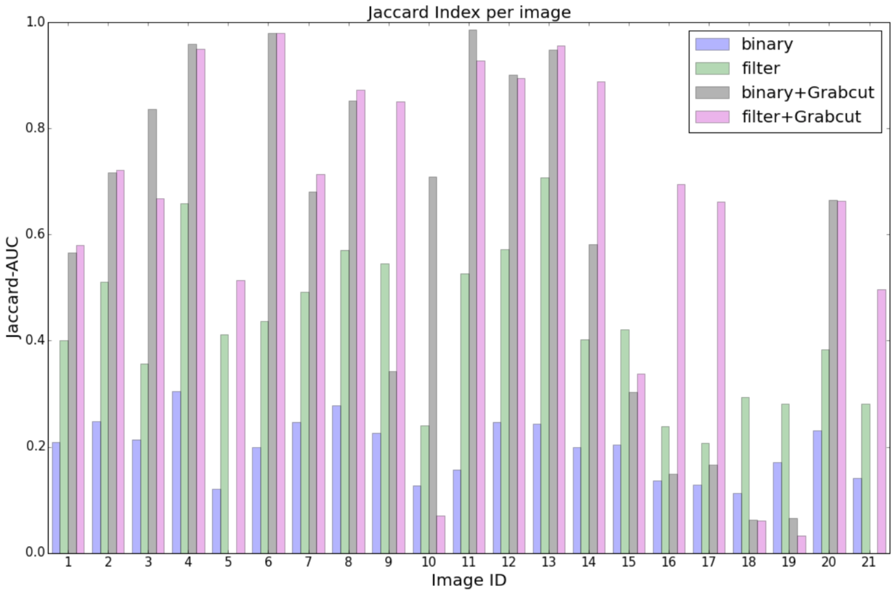

Fig. 11: Average Jaccard index across users per image

the algorithm. Nevertheless, for the same images, the *Filtered* versions represent a decent estimation of the object location and Grabcut still performing poorly. This fact evidences that the algorithm does not perform well when colors and textures of the foreground objects are similar to those composing the background, independently of the quality of the binary maps used as a seed for the interactive segmentation algorithm.

Table 6: Jaccard Index for images of Figure13

| imageID | *Binarized* | Binarized+Grabcut | *Filtered* | Filtered+Grabcut |
|---|---|---|---|---|
| 15 | .16 | .29 | .53 | .41 |
| 19 | .21 | .03 | .36 | .02 |
| 18 | .11 | .07 | .30 | .08 |
| 21 | .17 | .00 | .31 | .65 |
| 22 | .19 | .47 | .41 | .47 |

## 7.2 Combining EEG maps of different users

To reduce the noise of the EEG maps, we computed a single map per image by averaging the EEG maps of the different users. The parameters are set by

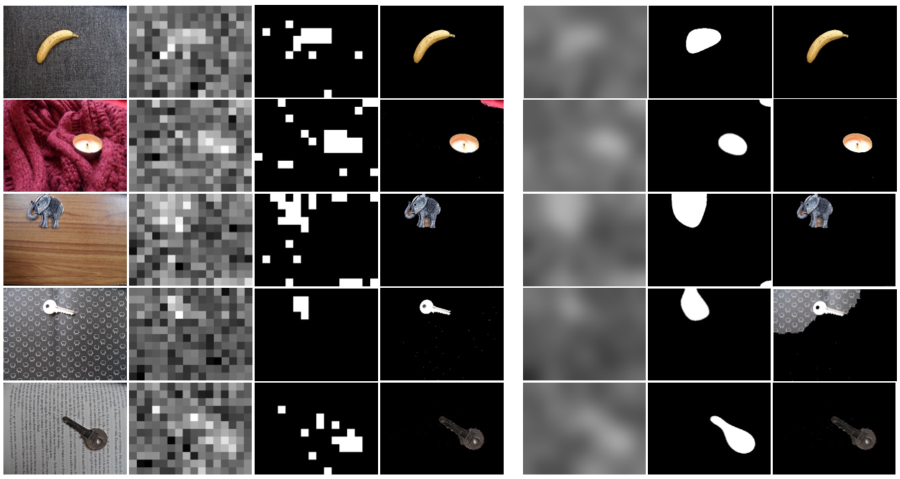

Fig. 12: Segmentations obtained when using Grabcut for user 3 in the second iteration of the crossvalidation. On the left, segmentations obtained by thresholding, on the right segmentations obtained by filtering and thresholding.

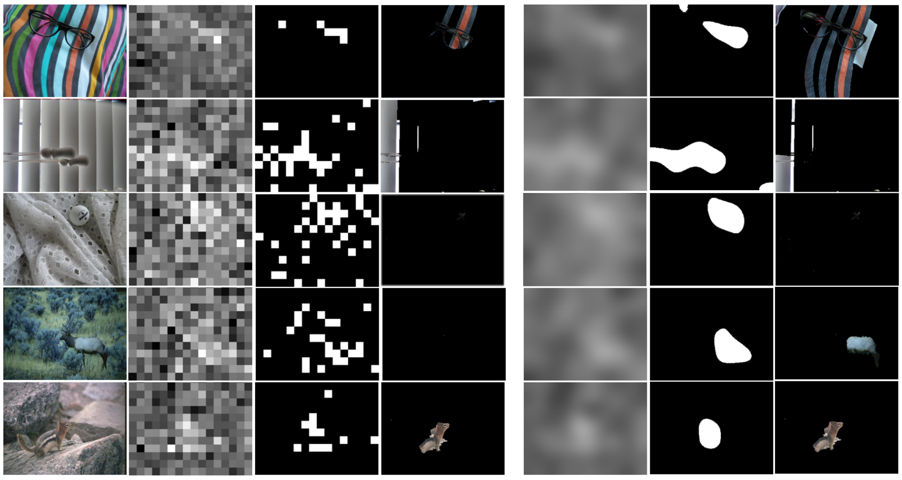

Fig. 13: Example of bad segmentations for user 3.

averaging across iterations and users. The threshold value for the binary EEG maps is 0.65 and for the filtered is 0.75 with gamma parameter of 22.52.

Qualitative results of the averaged EEG maps provide evidence that by combining the individual maps of different users it is possible to generate cleaner EEG maps (Figure 14). With the EEG maps averaged across users, we duplicate the quality of the binary maps: for the binary maps, we obtain a Jaccard value of $0.40 \pm 0.12$ versus the $0.20 \pm 0.10$ previously obtained. Also, for the filtered and binarized maps we obtain a Jaccard Indexes of $0.57 \pm$

0.12 versus the previous 0.43 ± 0.18. When adding Grabcut both versions perform similar, obtaining 0.70 ± 0.30 and 0.73 ± 0.27 for the binary and filtered masks versions respectively. These results are slightly better than the previous Grabcut results when processing users separately (0.55 ± 0.4 for the binary EEG maps and 0.66 ± 0.32 for the filtered EEG maps).

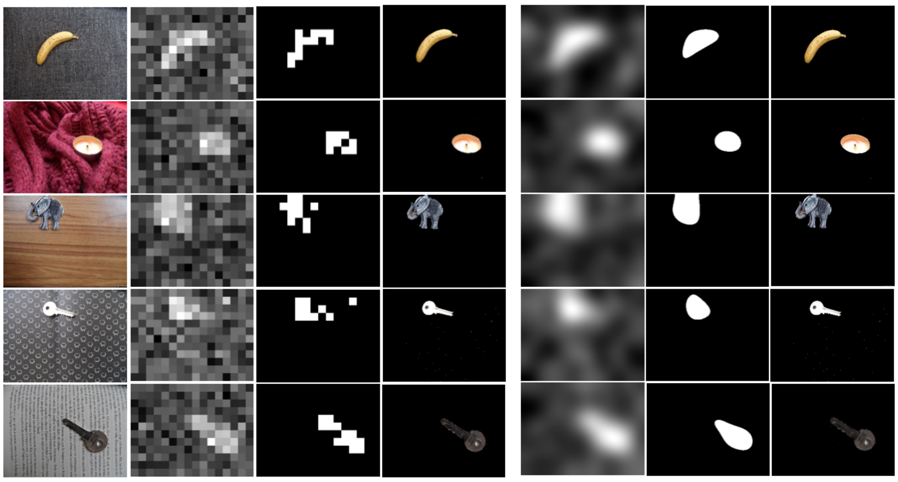

Fig. 14: Obtained results when using averaged EEG maps for user 3 in the second iteration of the system cross-validation.

## 8 Conclusions and future work

This work has presented and extended our previous publication [17] to segment objects from images by analysing EEG-signals in an attentional oriented task. Results indicate that a saliency map of the image can be estimated by partitioning it into windows and displaying them in a Rapid Serial Visual Presentation. The techniques presented in this paper have improved both the individual classification of the image windows and the final Jaccard index.

Firstly, by replacing the RBF kernel of the SVM classifier for a linear one, to improve the classification of the windows into containers or not containers of the object. This change already increased the mean classification ROC-AUC from 0.70 to 0.79.

Secondly, the initialization of the GrabCut segmentation algorithm has also been simplified. While three types of seeds (*definitely background*, *possible background* and *possible foreground*) were used in [17], now only two only labels (*possible background* and *possible foreground*) were considered. This changed combined with the linear SVM kernel, has increased the average Jaccard index from 0.47 to 0.66.

The paper also presented a detailed study showing that higher detection rates were obtained on windows containing an object occupation between $10\% - 70\%$. In addition, all results have also been carefully analysed by means of t-scores from statistical analysis.

This study offers a new communication opportunity to those patients affected by the Locked In Syndrome, which have no other way for interaction than brain activity. A segmentation system like this, even if imperfect, may allow these users to point at regions of interest in natural images such as their view field.

Future work should address more complex situations where multiple salient objects are present in the image to explore whether the simple *foregorund-background* pixel classification can also be extended to a multi-instance case. While it seems clear that the system can precisely locate the saliency parts of a natural image, it is still an open question if it could discriminate among them.

**Acknowledgements** This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under grant number SFI/12/RC/2289 and partially funded by the Project TEC2013-43935-R BigGraph of the Spanish Government.

## References

1. G. Bauer, F. Gerstenbrand, and E. Rumpl. Varieties of the locked-in syndrome. *Journal of Neurology*, 221(2):77–91, 1979.
2. C. J. Bell, P. Shenoy, R. Chalodhorn, and R. Rao. Control of a humanoid robot by a noninvasive brain computer interface in humans. *Journal of Neural Engineering*, 16(5):432–441, 2008.
3. N. Bigdely-Shamlo, A. Vankov, R. Ramirez, and S. Makeig. Brain activity-based image classification from rapid serial visual presentation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(5):432–441, 2008.
4. G. Bradski. *Dr. Dobb's Journal of Software Tools*, 2000.
5. D. Cruse, S. Chennu, C. Chatelle, T. A. Bekinschtein, D. Fernández-Espejo, J. D. Pickard, S. Laureys, and A. M. Owen. Bedside detection of awareness in the vegetative state: a cohort study. *The Lancet*, 378(9809):2088–2094, 2012.
6. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The Pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.
7. D. Fernandez-Canellas. Modeling the temporal dependency of brain responses to rapidly presented stimuli in erp based bci. Master's thesis, Northeastern University, 2013.
8. G. Healy and A. Smeaton. Eye fixation related potentials in a target search task. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 4203–4206, Aug 2011.
9. G. Healy and A. F. Smeaton. Optimising the number of channels in eeg-augmented image search. In *Proceedings of the 25th BCS Conference on Human-Computer Interaction*, BCS-HCI, pages 157–162, 2011.
10. R. Hebbalaguppe, K. McGuinness, J. Kuklyte, G. Healy, N. O. Connor, and A. Smeaton. How Interaction Methods Affect Image Segmentation : User Experience in the Task. In *Proc. The 1st IEEE Workshop on User-Centred Computer Vision (UCCV)*, 2013.
11. X. Hu, K. Li, J. Han, X. Hua, L. Guo, and T. Liu. Bridging the semantic gap via functional brain imaging. *Multimedia, IEEE Transactions on*, 14(2):314–325, 2012.

12. Y. Huang, D. Erdogmus, M. Pavel, S. Mathan, and K. E. Hild, II. A framework for rapid visual image search using single-trial brain evoked responses. *Neurocomputing*, 74(12-13):2041–2051, June 2011.
13. A. Kapoor, P. Shenoy, and D. Tan. Combining brain computer interfaces with vision for object categorization. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.
14. S. J. Luck. *An introduction to the event-related potential technique.* MIT Press, 2005.
15. R. Margolin, L. Zelnik-Manor, and A. Tal. How to evaluate foreground maps? In *CVPR*, 2014.
16. D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, volume 2, pages 416–423, July 2001.
17. E. Mohedano, G. Healy, K. McGuinness, X. Giró-i Nieto, N. E. O'Connor, and A. F. Smeaton. Object segmentation in images using eeg signals. In *Proceedings of the ACM International Conference on Multimedia*, MM '14, pages 417–426, New York, NY, USA, 2014. ACM.
18. S. Motomura, Y. Ojima, and N. Zhong. Eeg/erp meets act-r: A case study for investigating human computation mechanism. In N. Zhong, K. Li, S. Lu, and L. Chen, editors, *Brain Informatics*, volume 5819 of *Lecture Notes in Computer Science*, pages 63–73. 2009.
19. I. Pathirage, K. Khokar, E. Klay, R. Alqasemi, and R. Dubey. A vision based p300 brain computer interface for grasping using a wheelchair-mounted robotic arm. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 188–193, July 2013.
20. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
21. B. Roark, B. Oken, F.-O. M., U. Orhan, and D. Erdogmus. Offline analysis of context contribution to erp-based typing bci performance. *Journal of Neural Engineering*, 10(6):432–441, 2013.
22. C. Rother, V. Kolmogorov, and A. Blake. "GrabCut": Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, August 2004.
23. P. Sajda, E. Pohlmeyer, J. Wang, L. C. Parra, C. Christoforou, J. Dmochowski, B. Hanna, C. Bahlmann, M. K. Singh, and S.-F. Chang. In a blink of an eye and a switch of a transistor: cortically coupled computer vision. *Proceedings of the IEEE*, 98(3):462–478, 2010.
24. R. Spence. Rapid, Serial and Visual: a presentation technique with potential. *Information Visualization*, 1(1):13–19, 2002.
25. J. Wang, E. Pohlmeyer, B. Hanna, Y.-G. Jiang, P. Sajda, and S.-F. Chang. Brain state decoding for rapid image retrieval. In *Proceedings of the 17th ACM International Conference on Multimedia*, MM '09, pages 945–954, 2009.
26. A. Yazdani, J.-M. Vesin, D. Izzo, C. Ampatzis, and T. Ebrahimi. Implicit retrieval of salient images using brain computer interface. In *ICIP*, pages 3169–3172, 2010.