IDIAP RESEARCH REPORT

Detecting Intentional Mental Transitions in an Asynchronous BCI

Ferran Galán [a b] Francesc Oliva [c]
Joan Guàrdia [b] Pierre W. Ferrez [a]
José del R. Millán [a]

IDIAP–RR 06-43

January 2007

submitted for publication

[a] IDIAP Research Institute
[b] Dept. of Behavior Science Methods, University of Barcelona
[c] Dept. of Statistics, University of Barcelona

IDIAP Research Institute **www.idiap.ch**
Rue du Simplon 4 P.O. Box 592 1920 Martigny – Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 06-43

Detecting Intentional Mental Transitions in an Asynchronous BCI

Ferran Galán Francesc Oliva Joan Guàrdia Pierre W. Ferrez
José del R. Millán

January 2007

submitted for publication

Abstract. The inclusion of mental tasks transitions detection (MTTD) has proven a useful tool in guiding the transduction process of a BCI working under an asynchronous protocol. MTTD allows for the extraction of the signal's contextual information in order to infer the user's intentionality at a given moment and thus correcting possible classification errors. Despite the good results shown, the algorithm previously proposed [1] does not show good behavior in contexts where the user gets online feedback. The algorithm that we propose in this paper, like its antecessor, is based on canonical variates transformation (CVT) and on distance-based discriminant analysis (DBDA), but it has a new transitions detector based on Kalman filtering. In addition, it includes a classifier supervisor based on heuristics rules that exploit transition detection as well as inconsistencies between subject's mental intention and the associated EEG. These heuristic rules lead to significant improvements of the BCI in terms of both classification accuracy and channel capacity, adapting itself to the user's needs.

I. INTRODUCTION

The possibility of acting upon the environment that surrounds us without using our nervous system's efferent pathway opens the door to new ways of interaction that can compensate for total or partial loss of mobility. Research in brain-computer interface (BCI) has made it possible to transform brain activity into mental commands for navigating in virtual environments [2], controlling prosthetic devices [3], steering mobile robots [4] and wheelchairs, or writing on virtual keyboards [5], [6], [7]. BCI systems can be characterized along two orthogonal dimensions. The first one is the nature of the recorded brain signals, either invasive (obtained from implanted microelectrodes) or non-invasive (based on electroencephalogram, EEG, measured from scalp electrodes). The second dimension is the interaction protocol, either synchronous (time-locked to externally-paced cues) or asynchronous (guided by the user's self-paced decisions). Our work is focused on asynchronous non-invasive BCIs so that humans can interact with brain-actuated devices as flexibly and naturally as possible.

The algorithm proposed in this paper is an extension of our previous work [1], the winner of the last international BCI Competition III for 'Data set V' [8]. The new algorithm, like its antecessor, is based on canonical variates transformation (CVT) [9] and on distance-based discriminant analysis (DBDA) [10], but it has a new transitions detector based on Kalman filtering. In addition, it includes a classifier supervisor based on heuristics rules that exploit transition detection as well as inconsistencies between subject's mental intent and the associated EEG. These heuristic rules lead to significant improvements of the BCI in terms of both classification accuracy and channel capacity.

II. METHODS

A. Data Acquisition and Mental Tasks

For the offline assessment of the proposed algorithm we have used two datasets. The first one is the 'Data set V' of the BCI Competition III [8], where subjects did not receive any feedback indicating their performance. The second one comes from feedback experiments. In both datasets, EEG signals were recorded with a portable Biosemi acquisition system from 32 (first dataset) or 64 (second dataset) electrodes. The sampling frequency was 512 Hz. The signal was spatially filtered using a surface Laplacian (first dataset) or common average reference (CAR) (second dataset) previous to the estimation every 62.5 ms. (16 times per second) of the PSD in the band 8-30-Hz, with a resolution of 2Hz over the last 1-second windows. In the first dataset the PSD was estimated on the electrodes C3, Cz, C4, CP1, CP2, P3, Pz, P4; thus obtaining a 96-dimensional vector (8 electrodes $\times$ 12 frequency components) as a pattern. In the second dataset the PSD was estimated on F3, F4, FC1, FC2, C3, C1, C2, C4, CP1 and CP2 obtaining a 120-dimensional vector (10 electrodes $\times$ 12 frequency components). The computation of the EEG patterns, the vector **PSD**, is described in [8] and [11].

Data come from 5 healthy voluntary subjects. Subjects 1, 2 and 3 from the first dataset carried out three mental tasks, two motor imagination tasks (right-left hand movement imagination) and one cognitive task (search of words with the same initial letter) for 15 seconds switching randomly between them at the operator's request. EEG signals were recorded during 4 non-feedback sessions (see [10] for details). Subjects 4 and 5 executed the same mental tasks also during 4 sessions while receiving continuous visual feedback from the movement of a virtual robot in the case of subject 4, or from a real mobile robot in the case of subject 5. The robots were controlled by the association between the subject's mental states and high-level commands (right-left hand imagination with right-left turns, and word search with forward movements). In addition, the brain-actuated robots relied on a behavior-based controller that guarantee obstacle avoidance and smoothed turns [4]. These two subjects were at the very beginning of their training with the robots.

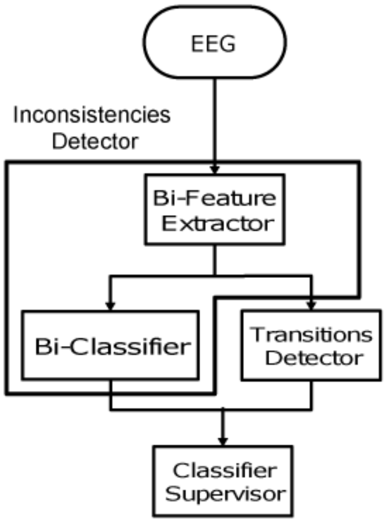

Figure 1: Schematic representation of the whole algorithm. Note that the inconsistencies detector (based on a bi-feature extractor and a bi-classifier) and the transitions detector work in parallel. The output of the two previous components is used by the classifier supervisor.

For every subject, the first three sessions were used as training and validation sets while the fourth session was the test set.

B. Algorithm Description

The proposed algorithm incorporates three components: an *inconsistencies detector* (Section II.C) between the user's intent and the measured EEG patterns that assigns a provisional label to each incoming pattern, a *transitions detector* based on Kalman filtering (Section II.D) that detects patterns not similar to their predecessors, and a *classifier supervisor* based on three heuristic rules (Section II.E) that determine which of these patterns identified as transitions by the transitions detector are intentional transitions between different mental tasks. Figure 1 depicts a schematic representation of the overall algorithm. The inconsistencies detector, based on a bi-feature extractor and a bi-classifier, works in parallel with the transition detector. Thus, each incoming pattern is first checked for inconsistency and transition. Next, based on three heuristic rules and only using the labels of the three last patterns identified as transitions, the classifier supervisor eventually changes the label of the pattern provided by the inconsistencies detector.

C. Inconsistencies Detector

One of the main difficulties of classifying spontaneous brain activity is its variability over time. Physiological reasons, like endogenous brain processes and fatigue, cause slow changes of EEG over time. An approach to deal with this type of non-stationarity is to build adaptive classifiers that track this variability [11]. In addition, rapid shifts in the user's motivational and attentional states[1] may lead to a mismatch between the user's intent and the user's EEG patterns. The aim of the inconsistencies detector is to identify those patterns that likely are inconsistent with the user's intent.

In order to identify inconsistent patterns we propose to build two canonical spaces by means of a canonical variates transformation (CVT), as in [1]. The first one is based on the original labelling

[1]For instance, erroneous responses of the BCI, especially if they are frequent as it is the case at the beginning of the user's training, may disconcert and frustrate the user.

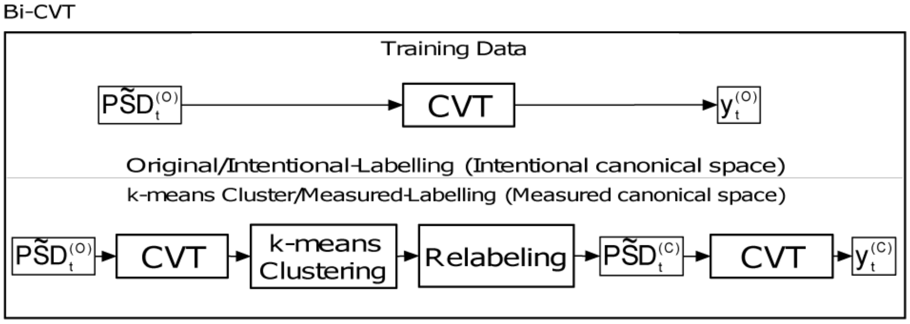

Figure 2: Bi-feature extractor based on CVT (Bi-CVT). From the training data we build two canonical spaces, the intentional canonical space using the original labelling (top), and the measured canonical space using the k-means cluster labelling (bottom). Patterns whose projections onto the canonical spaces do not match are potentially inconsistent and the Bi-DBDA labels them as *unknown*.

that corresponds to the user's intent (*intentional* labelling)—i.e., it is obtained by a direct CVT on the training set. In the second case, after building the canonical space as in the first case, we carry out k-means cluster analysis (using Euclidean distance) and generate as many clusters as mental tasks (3, in our case). Then we relabel the patterns in the training set with the labels obtained from the clustering process (*measured* labelling) and build a new canonical space. This Bi-CVT process yields two different canonical spaces, the intentional and the measured canonical spaces (see Figure 2). Afterwards, we project the patterns of the test set onto both canonical spaces and the resulting projections are sent to each DBDA classifier. The outputs of these two classifiers are combined to obtain a final decision, called Bi-DBDA, which labels a pattern as *unknown* if the outputs of the two previous DBDA classifiers are inconsistent—i.e, they are different.

The different steps of the inconsistencies detector are the following[2]:

1. *Normalization:* Each spectral component h of channel i from the pattern recorded at time t, $PSD_{h_t}(i)$, is normalized by the energy of the channel

$$\widetilde{PSD}_{h_t}(i) = \frac{PSD_{h_t}(i)}{\sum_{h=1}^{n} PSD_{h_t}(i)} \tag{1}$$

2. *Bi-CVT:* We find the eigenvectors $\mathbf{A}^{(O)}$ and $\mathbf{A}^{(C)}$ of $\mathbf{W}^{-1(O)}\mathbf{B}^{(O)}$ and $\mathbf{W}^{-1(C)}\mathbf{B}^{(C)}$ with eigenvalues greater than zero, where $\mathbf{B}^{(\cdot)}$ and $\mathbf{W}^{(\cdot)}$ are the between- and within-classes dispersion matrices, obtained from the matrices $\widetilde{\mathbf{PSD}}^{(O)}$ and $\widetilde{\mathbf{PSD}}^{(C)}$ of patterns with the original labels (*intentional* labelling) or with the cluster-analysis labels (*measured* labelling), respectively. Then, we compute the projections

$$\mathbf{y}_t^{(O)} = \widetilde{\mathbf{PSD}}_t^{(O)} \ \mathbf{A}^{(O)} \tag{2}$$

$$\mathbf{y}_t^{(C)} = \widetilde{\mathbf{PSD}}_t^{(C)} \ \mathbf{A}^{(C)} \tag{3}$$

3. *Bi-DBDA:* The label ν_t of the incoming pattern at time t with projections $\mathbf{y}_t^{(O)}$ and $\mathbf{y}_t^{(C)}$ is

$$\nu_t = \begin{cases} g & \text{if } (\overline{\phi}_g(\mathbf{y}_t^{(O)}) = min_k[\overline{\phi}_k(\mathbf{y}_t^{(O)})]) \bigcap (\overline{\phi}_g(\mathbf{y}_t^{(C)}) = min_k[\overline{\phi}_k(\mathbf{y}_t^{(C)})]) \\ unknown & \text{otherwise} \end{cases} \tag{4}$$

[2]Note that the first step is the normalization of the frequency components of each electrode.

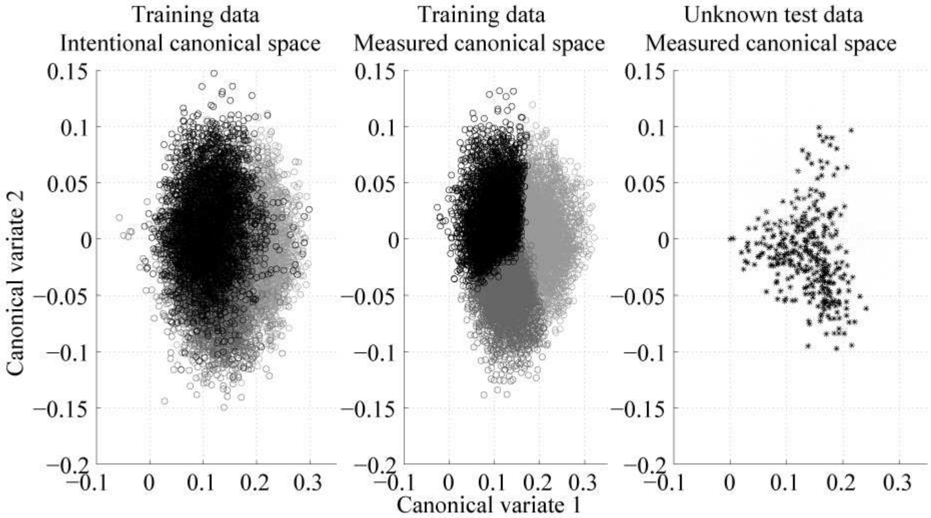

Figure 3: Bi-DBDA example, subject 1. *Left*: Training data projection with original labels on the intentional canonical space (black: imagination of left hand movement; dark grey: imagination of left hand movement, light grey: word search). *Middle*: Training data projection with cluster labels on the measured canonical space. *Right*: Test data labelled by Bi-DBDA as unknown projected on the measured canonical space. These inconsistent patterns are those projected onto the mismatch areas of the two canonical spaces.

where k is the number of classes and $\overline{\phi}_g(\mathbf{y}^{(\cdot)})$ is the average of DBDA proximity estimates, $\hat{\phi}_g(\mathbf{y}^{(\cdot)})$ (see appendix), in the corresponding canonical space over $N_{av} = 4$ consecutive patterns

$$\overline{\phi}_g(\mathbf{y}_t^{(\cdot)}) = \frac{1}{N_{av}} \sum_{i=1}^{N_{av}} \hat{\phi}_g(\mathbf{y}_{t-i+1}^{(\cdot)}) \tag{5}$$

Thus, the final decision is obtained every 0.250 s. In this way, the new pattern is assigned to class C_g only if the two DBDA classifiers agree. See Figure 3 for an example.

D. Transitions Detector Based on Kalman Filtering

Kalman filtering is a principled approach to detect abrupt changes in temporal series [12]. We have used it to build a more robust transitions detector than in the previous algorithm [1]. While the inconsistencies detector filters patterns given the relative positions of its projections on both training canonical spaces, the transitions detector checks for patterns whose distances to their predecessors on the measured canonical space are larger than expected. Thus, the transitions detector filters those patterns that are likely intentional transitions between different mental tasks.

Using the projected patterns ($\mathbf{y}_t^{(C)}$) in the canonical space obtained with labels computed by k-means cluster analysis (*measured* labelling), we have built the linear state dynamical system

$$\begin{aligned} \mathbf{x}_t &= \mathbf{A}\mathbf{x}_{t-1} + \mathbf{w}_{t-1} \\ \mathbf{y}_{t-1}^{(C)} &= \mathbf{H}\mathbf{x}_{t-1} + \mathbf{v}_{t-1} \end{aligned} \tag{6}$$

where $\mathbf{x}_t$, $\mathbf{A}$ and $\mathbf{H}$ are the state vector, the state matrix and the measurement matrix, respectively. The state noise $\mathbf{w}_{t-1}$ and the measurement noise $\mathbf{v}_{t-1}$ are assumed to be uncorrelated zero-mean

Gaussian white-noise processes with covariance matrices $\mathbf{Q}$ and $\mathbf{R}$. The Kalman filter finds the optimal state estimate $\hat{\mathbf{x}}_t$ (minimazing the variance error estimator, defined as $E[\|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2]$) given a set of past observations $\{\mathbf{y}_1^{(C)},...,\mathbf{y}_t^{(C)}\}$ in a prediction-correction approach:

1. **Prediction equations**. In this step, the prediction of the state of the system at time t $\hat{\mathbf{x}}_{t|t-1}$ and his variance $\mathbf{P}_{t|t-1}$ are computed from the estimates and $\mathbf{P}_{t-1|t-1}$ at time *t-1* and the noise covariance $\mathbf{Q}$:

$$\hat{\mathbf{x}}_{t|t-1} = \mathbf{A}\hat{\mathbf{x}}_{t-1|t-1} \tag{7}$$

$$\mathbf{P}_{t|t-1} = \mathbf{A}\mathbf{P}_{t-1|t-1}\mathbf{A}^T + \mathbf{Q} \tag{8}$$

   Initial matrices $\mathbf{A}$, $\mathbf{H}$, $\mathbf{Q}$ and $\mathbf{R}$ were estimated with EM algorithm [13] using the third session. As initial values estimates we have fixed $\hat{\mathbf{x}}_{t-1|t-1} = \mathbf{y}_{t-1}^{(C)}$ and $\mathbf{P}_{t-1|t-1} = \mathbf{R}$.

2. **Correction equations**. The obtained estimates in the prediction step are corrected by the innovation process

$$\epsilon_t = \mathbf{y}_t^{(C)} - \mathbf{H}\hat{\mathbf{x}}_{t|t-1} \tag{9}$$

   defined as the difference between the new measured value $\mathbf{y}_t^{(C)}$ and the hypothetical measured value given the estimate $\hat{\mathbf{x}}_{t|t-1}$ at time t. Then, the corrected estimates are

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t[\mathbf{y}_t^{(C)} - \mathbf{H}\hat{\mathbf{x}}_{t|t-1}] \tag{10}$$

$$\mathbf{P}_{t|t} = [\mathbf{I} - \mathbf{K}_t\mathbf{H}]\mathbf{P}_{t|t-1} \tag{11}$$

   where the Kalman gain matrix is

$$\mathbf{K}_t = \mathbf{P}_{t|t-1}\mathbf{H}^T\mathbf{S}_t^{-1} \tag{12}$$

   and the innovation covariance matrix is

$$\mathbf{S}_t = \mathbf{H}\mathbf{P}_{t|t-1}\mathbf{H}^T + \mathbf{R} \tag{13}$$

If the system works properly, the normalized innovation process $\tilde{\epsilon_t} = \mathbf{S}_t^{-1/2}\epsilon_t$ is a zero-mean Gaussian white-noise process with identity covariance matrix $E[\tilde{\epsilon_t}] = 0$, $E[\tilde{\epsilon_t}\tilde{\epsilon_t}^T] = \mathbf{I}$. Thus, any transition or variation in the model is reflected by a change in the aforementioned statistics.
In order to detect transitions we use the sequence of innovation process sample covariance matrices [14]

$$\mathbf{U}_t = \frac{1}{N_{av}-1}\sum_{i=t-N_{av}+1}^{t}[\tilde{\epsilon_i} - \bar{\tilde{\epsilon_t}}][\tilde{\epsilon_i} - \bar{\tilde{\epsilon_t}}]^T \tag{14}$$

where

$$\bar{\tilde{\epsilon_t}} = \frac{1}{N_{av}}\sum_{i=t-N_{av}+1}^{t}\tilde{\epsilon_i} \tag{15}$$

and $N_{av} = 2$ consecutive patterns. Given a threshold θ, we consider that there is a transition if the following inequality is satisfied

$$I(\mathbf{U}_t) > \theta > I(\mathbf{U}_{t-1}) \tag{16}$$

where

$$I(\mathbf{U}_t) = \Psi[d(\mathbf{U}_t), d(\mathbf{U}_{t-1})] \tag{17}$$

is the Euclidean distance $\Psi[\cdot]$ between the diagonal vectors $d(\mathbf{U}_t)$ and $d(\mathbf{U}_{t-1})$ (variance vectors) of two consecutive sample covariance matrices. In this way, using (16) we only pay attention to abrupt changes in time. Consecutive changes are not considered to be intentional—rather they may indicate periods where the subject cannot sustain attention.

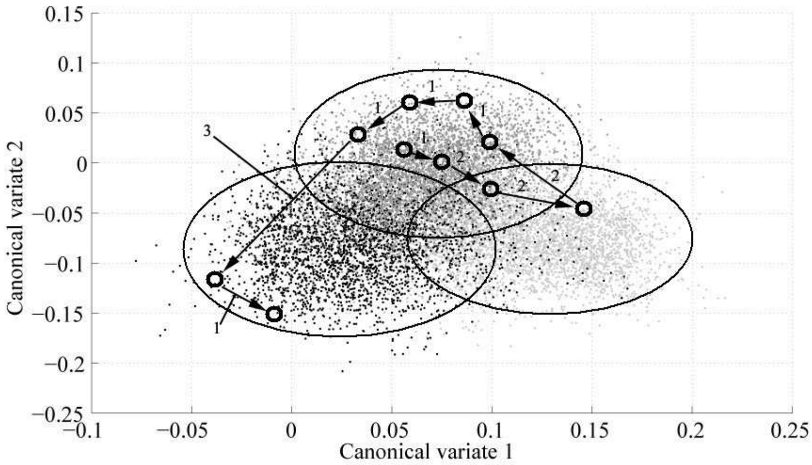

Figure 4: Examples of projected test patterns identified as transitions on the training measured canonical space (black: imagination of left hand movement; dark grey: imagination of right hand movement; light grey: word search). Dots show the projected training patterns. Big circles represent the location of the three mental tasks in the canonical space. Small circle indicate transitions; i.e., patterns exhibiting leaps in the measured canonical space with respect to their predecessors. In this example the user executes first imagination of right hand movement and then imagination of left hand movement. This example illustrates the three kinds of transitions: **1**, within-class; **2**, unintentional between-class; **3**, intentional between-class transition.

E. Classifier Supervisor

As described in Section II.D incoming patterns identified as transitions exhibit leaps in the training measured canonical space with respect to their predecessors. Thus these transitions patterns are probably located in different canonical subspaces from their predecessors and Bi-DBDA will have labelled such a transition pattern either as the majority of its predecessors (*within-class transition*) or differently (*intentional* or *unintentional between-class transition*) (see Figure 4). This last scenario reflects a mismatch between the user's intent and the user's EEG patterns. The goal of the classifier supervisor is to infer the different kinds of transitions and correct the labelling produced by Bi-DBDA.

To do so we have designed three heuristic rules (HR1, HR2, and HR3). HR1 is the simplest and most conservative rule as it rejects a large number of patterns (labels them as *unknown*) rule from which HR2 and HR3 are derived (see Figures 5 and 6 for a detailed description). The three HR label a new pattern, always every 0.250 s, using information from the last three transitions. The three HR seek to infer the intentional label of the patterns identified as transitions and then reclassify the subsequent patterns, until the next transition, with the same label. In this way the classifier supervisor only changes the labels when a intentional transition between different mental tasks is inferred.

III. RESULTS AND DISCUSSION

The advantages of a inconsistencies detector and a classifier supervisor in an asynchronous BCI have been assessed offline by using the first three sessions of each subject as training and validation[3] data

[3]k-fold cross-validation was done to select the values of the different hyperparameters of the algorithm—e.g., thresholds of the heuristics rules.

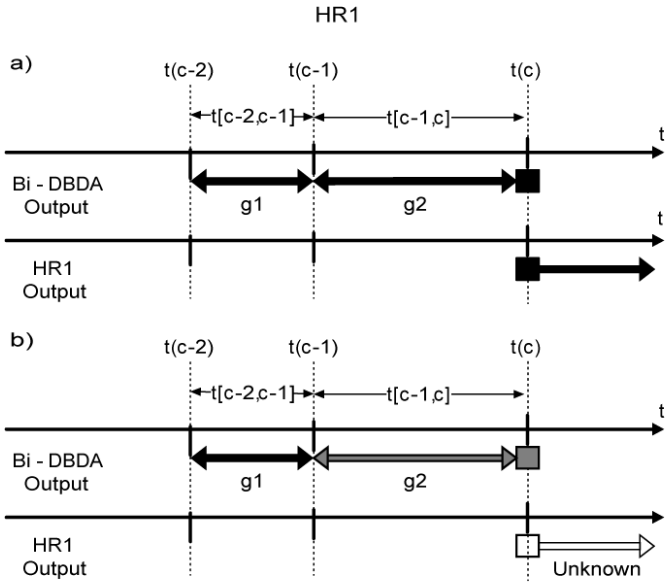

Figure 5: HR1 changes the label of all *unknown* patterns after a transition if the C_{g1} and C_{g2} classes with maximum labelling proportion assigned by Bi-DBDA, $max_{k+1}[p(\nu_{t[c-1,c]} = g1)])$ and $max_{k+1}[p(\nu_{t[c-2,c-1]} = g2)])$, in the two gaps $t[c-2, c-1]$, $t[c-1, c]$ limited by the last three transitions are the same and equal to the label of the last transition t(c) (**HR1.a**). Otherwise, HR1 labels as *unknown* the last transition t(c) and all following patterns (**HR1.b**). Note that the number of classes $k+1$ corresponds to the k mental tasks plus the *unknown* class. Although this labelling rule may delay the detection of an intentional transition, it allows for the filtering of a great number of unintentional transitions. This rule yields a large number of unknown patterns, which limits its suitability to situations where it is useful to be conservative (e.g., at the very early stages of training where there is a higher degree of mismatch between the user's intent and the EEG patterns).Empty squares referred to unknown labels.

and the fourth as test data. Performance has been measured in terms of classification accuracy and channel capacity. Given that the different components of the algorithm may reject responses (*unknown* responses), the estimator proposed by Millán et al. [4] has been used as a measure of the channel capacity. Table 1 shows the results for subjects 1, 2 and 3, whose data were used in BCI Competition III. Table 2 shows the results obtained on the subjects who received online feedback.

Regarding the first dataset, all the components of the new algorithm give the best performance for subject 1 and the worst for subject 3, paralleling the results obtained by the different algorithms that participated in the BCI Competition III. $DBDA^{(O)}$ classifier (DBDA trained with the intentional canonical space) and $DBDA^{(C)}$ classifier (DBDA trained with measured canonical space) exhibit a very similar performance over the three subjects in terms of both channel capacity and classification accuracy. Note that although the classification accuracy of these simple classifiers are slightly worse than that of the original classifier BCI-III, the channel capacity is similar (or even better) due to the fact that the new algorithm yields faster responses. But the real advantage of the new algorithm appears when the two new components process sequentially the outputs of the two DBDA classifiers. Indeed, the detection of inconsistent patterns (7.82%, 12.18% and 30.75% unknown responses for subjects 1, 2 and 3, respectively) leads to a significant increase of the Bi-DBDA channel capacity since the percentage of patterns incorrectly classified is greatly reduced. In particular, the effects of the inconsistencies detection in subject 3 stands out: the percentages of correctly and incorrectly classified patterns are inverted.

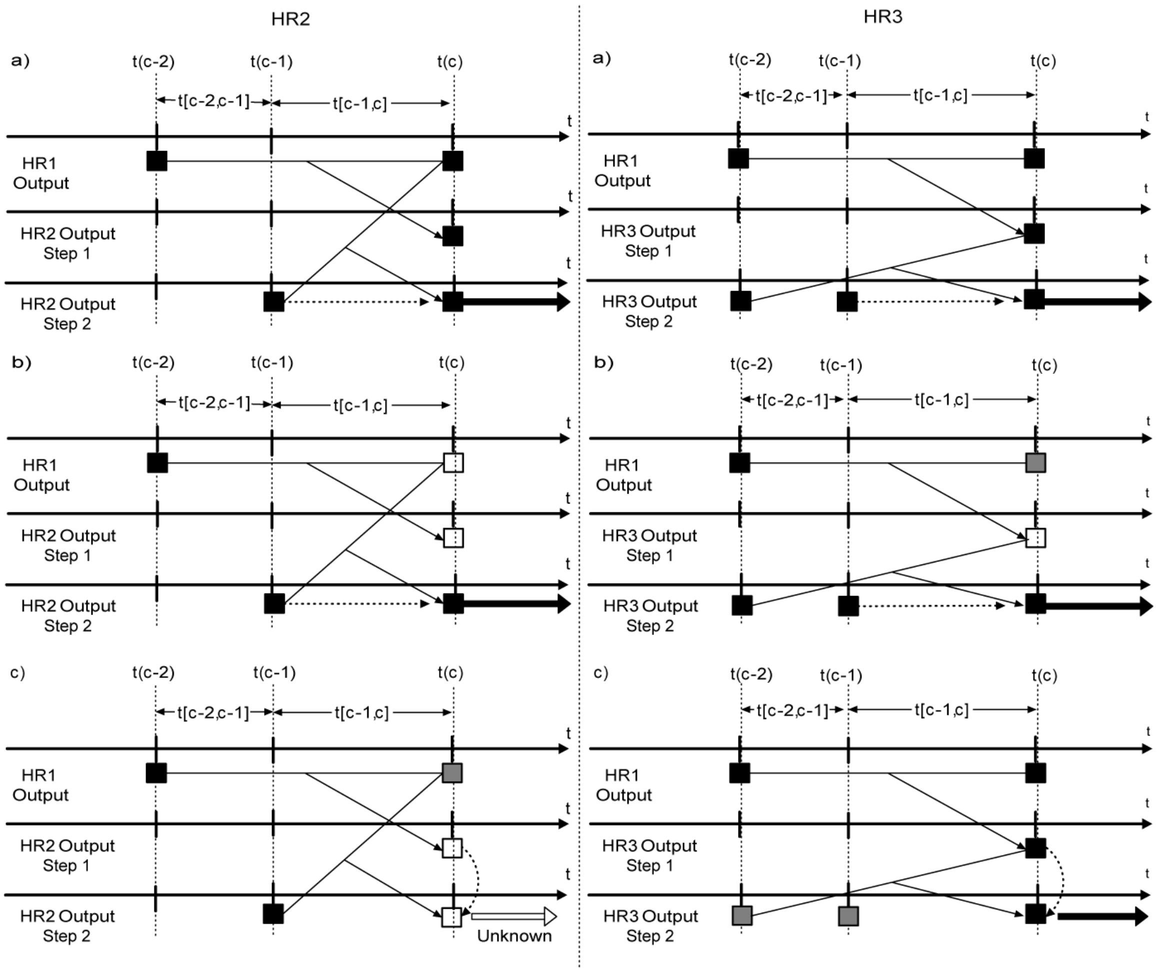

Figure 6: Rules HR2 and HR3 incorporate two different ways to extend the filtering of unintentional transitions generated by HR1, allowing a more extensive supervision with different degrees of caution. Both of them share the first step where the new transition t(c) keeps the label it received from HR1 if it is the same as the label HR1 assigned to transition t(c-2), otherwise t(c) is classified as *unknown*. HR2 and HR3 differ in the second step. HR2 re-labels the new transition t(c) as the previous one t(c-1) if HR1 labelled t(c) either as HR2 labelled t(c-1) (**HR2.a**) or as *unknown* (**HR2.b**), otherwise the output of the first step of HR2 is kept (**HR2.c**). HR3 re-labels the new transition t(c) as the previous one t(c-1) if HR3 in the first step labelled t(c) either as HR3 labelled t(c-2) in the second step (**HR3.a**) or as *unknown* (**HR3.b**), otherwise the output of the first step of HR3 is kept (**HR3.a**). The difference between both rules is the amount of *unknown* labels generated: HR3 rejects a shorter number of patterns. Empty squares referred to unknown labels.

Table 1: Classifiers performance on the BCI Competition III dataset.

Subject1

Classifier	Channel Capacity	Accuracy	Error	Rejection
$DBDA^{(O)}$	1.69 b/s	70.80 %	29.20 %	-
$DBDA^{(C)}$	1.68 b/s	70.72 %	29.28 %	-
Bi-DBDA	1.96 b/s	67.47 %	24.71 %	7.82 %
HR1	2.90 b/s	54.02 %	5.61 %	40.37 %
HR2	3.91 b/s	72.37 %	4.60 %	23.03 %
HR3	4.34 b/s	90.77 %	8.14 %	1.09 %
BCI-III	1.30 b/s	79.60 %	20.40 %	-

Subject2

Classifier	Channel Capacity	Accuracy	Error	Rejection
$DBDA^{(O)}$	.85 b/s	59.92 %	40.08 %	-
$DBDA^{(C)}$	.82 b/s	59.44 %	40.56 %	-
Bi-DBDA	1.17 b/s	54.51 %	33.31 %	12.18 %
HR1	1.89 b/s	38.24 %	11.84 %	49.92 %
HR2	2.68 b/s	58.31 %	11.31 %	30.38 %
HR3	2.75 b/s	80.12 %	19.04 %	.84 %
BCI-III	.82 b/s	70.31 %	29.69 %	-

Subject3

Classifier	Channel Capacity	Accuracy	Error	Rejection
$DBDA^{(O)}$	.30 b/s	48.74 %	51.16 %	-
$DBDA^{(C)}$	.30 b/s	48.77 %	51.13 %	-
Bi-DBDA	.98 b/s	37.00 %	32.25 %	30.75 %
HR1	1.46 b/s	28.78 %	14.52 %	56.70 %
HR2	1.71 b/s	41.81 %	18.49 %	39.70 %
HR3	.49 b/s	52.52 %	46.46 %	1.02 %
BCI-III	.31 b/s	56.02 %	43.98 %	-

$DBDA^{(O)}$: DBDA trained with training data, original labelling.
$DBDA^{(C)}$: DBDA trained with training data, cluster labelling.
BCI-III: Winner classifier of the BCI Competition III, which yields a decision every 0.5 s.
Note that the new algorithm yields a response every 0.250 s, hence achieving a higher channel capacity than the original algorithm for similar classification accuracies.

Table 2: Classifiers performance of subjects receiving online feedback.

Subject4

Classifier	Channel Capacity	Accuracy	Error	Rejection
$DBDA^{(O)}$	.11 b/s	42.73 %	57.27 %	-
$DBDA^{(C)}$	.10 b/s	42.30 %	57.70 %	-
Bi-DBDA	.22 b/s	40.00 %	52.89 %	7.11 %
HR1	1.10 b/s	17.95 %	13.75 %	68.30 %
HR2	1.48 b/s	26.57 %	10.41 %	63.02 %
HR3	.55 b/s	54.40 %	45.29 %	.31 %
BCI-III*	.06 b/s	43.75 %	56.25 %	-

Subject5

Classifier	Channel Capacity	Accuracy	Error	Rejection
$DBDA^{(O)}$	.04 b/s	39.04 %	60.96 %	-
$DBDA^{(C)}$	.04 b/s	39.06 %	60.94 %	-
Bi-DBDA	.12 b/s	36.47 %	56.46 %	7.07 %
HR1	.92 b/s	14.25 %	19.11 %	66.64 %
HR2	.85 b/s	18.14 %	27.02 %	54.84 %
HR3	.07 b/s	40.51 %	59.26 %	.23 %
BCI-III*	.02 b/s	39.75 %	60.25 %	-

$DBDA^{(O)}$: DBDA trained with training data, original labelling.
$DBDA^{(C)}$: DBDA trained with training data, cluster labelling.
BCI-III: Winner classifier of the BCI Competition III, which yields a decision every 0.5 s.
Note that the new algorithm yields a response every 0.250 s, hence achieving a higher channel capacity than the original algorithm for similar classification accuracies.

These results are further improved with the use of the different heuristic rules (HR) of the classifier supervisor. HR1 allows to increase the channel capacity for the three subjects, even though it rejects an extremely high percentage of patterns. HR2 rejects much less patterns and improves significantly the channel capacity. Finally, HR3 achieves a still further significant increase of the channel capacity (due to a remarkable increase in classification accuracy) for the first two subjects, but it shows a dramatic decline in performance in terms of channel capacity for subject 3.

Concerning the second dataset where subjects received continuous feedback, we observed the same trend; i.e., the first two DBDA classifiers already achieved similar classification accuracy than the original classifier (and, hence, better channel capacity due to their faster responses), and the channel capacity is further improved through the sequential processing of the outputs of those classifiers by the inconsistencies detector (Bi-DBDA) and classifier supervisor. However, the effects of the different HR on the subjects' performance is not the same as before. For subject 4 we also see a significant improvement after the application of HR1 on the output of the Bi-DBDA and even a higher one with HR2. HR3 also increases the channel capacity with respect to Bi-DBDA, but in a lesser extent than HR1 and HR2. Significantly, all three heuristic rules invert the percentages of correctly and incorrectly classified patterns. For subject 5, only HR1 and HR2 increase the channel capacity with respect to Bi-DBDA, with HR1 outperforming HR2. The disadvantage of HR3 for these two subjects is that it rejects a very short number of patterns and, given that these two subjects are at the very beginning of their training, the BCI makes risky decisions, thus generating a high percentage of misclassifications. Under this condition, more conservative heuristic rules are better suited.

In conclusion, the incorporation of the inconsistencies detector and the classifier supervisor outperforms the original classifier, winner of the BCI Competition III, on all five subjects for both working conditions of a BCI—namely with or without online feedback. Of the three heuristic rules, HR3 only seems suitable when the user's performance is already satisfactory (i.e., subjects 1 and 2). Otherwise, it is recommendable to use HR1 or, preferably, HR2. This is the case of subject 3, who has still a poor performance, and of subjects 4 and 5, who are at the beginning of their training or do not yet master the complexity of the online feedback coming from moving robots. Although HR1 and HR2 present a lower performance in terms of classification accuracy than HR3, they succeed notably in limiting the number of incorrect responses, an important aspect in order to avoid processes of discouragement, confusion and frustration that can easily interfere with attentional processes, and thus achieving acceptable levels of channel capacity.

IV. CONCLUSION

In this article we have shown that the inclusion of a inconsistencies detector and a classifier supervisor based on intentional mental transitions detection leads to an effective inference of the user's intended mental task. This approach yields a significant increase of the channel capacity mainly because it allows to decrease the error rates. Experimental results show the benefits of our algorithm in both working conditions of a BCI, namely with or without online feedback. The main limitation of our approach is the use of ad-hoc heuristic rules. The next step is to formalize those heuristic rules in a Bayesian framework and build probabilistic models for the inference of the user'intent as in [15].

V. ACKNOWLEDGEMENT

The authors would like to thank Ricardo Chavarriaga for his useful suggestions and Anna Buttfield for her collaboration in the experimental recordings.
This work is supported by the Ministry of Education and Universities of Catalonia, by the Swiss

National Science Foundation through the National Centre of Research on "Interactive Multimodal Information Management (IM2)" and by the European IST Programme FET Project FP6-003758. This paper only reflects the author's views and founding agencies are not liable for any use that may be made of the information contained herein.

VI. APPENDIX

As described in [10][1], given k classes $C_1, ..., C_k$ and a defined distance function d_g for class C_g, the proximity measurement for pattern w_t with vector $\mathbf{y}_t = \mathbf{y}(w_t)$, is defined as

$$\phi_g(\mathbf{y}_t) \quad = \quad V_g(\mathbf{y}_{C_g}|\mathbf{y}_t) - V_g(\mathbf{y}_{C_g}) \tag{18}$$

where

$$V_g(\mathbf{y}_{C_g}) \quad = \quad \frac{1}{2}E_{C_gC_g}[d_g^2(\mathbf{y}_{C_g},\mathbf{y}_{C_g})] \tag{19}$$

and

$$V_g(\mathbf{y}_{C_g}|\mathbf{y}_t) \quad = \quad E_{C_g}[d_g^2(\mathbf{y}_t,\mathbf{y}_{C_g})] \tag{20}$$

are the geometric variability and the relative geometric variability to pattern $\mathbf{y}_t$. Thus, the DBDA assigns w_t to C_g, if

$$\phi_g(\mathbf{y}_t) \quad = \quad min_k[\phi_k(\mathbf{y}_t)] \tag{21}$$

In practice, suitable estimates of the geometric variability and the relative geometric variability to pattern $\mathbf{y}_t$ are

$$\hat{V}_g(\mathbf{y}) \quad = \quad \frac{1}{2n_g^2}\sum_{j,j'=1}^{n_g} d^2(\mathbf{y}_{gj},\mathbf{y}_{gj'}) \tag{22}$$

$$\hat{V}_g(\mathbf{y}|\mathbf{y}_t) \quad = \quad \frac{1}{n_g}\sum_{j=1}^{n_g} d^2(\mathbf{y}_t,\mathbf{y}_{gj}), \tag{23}$$

where n_g is the number of patterns of class C_g and $\mathbf{y}_{gj}$ is the pattern j of class g. Therefore, the estimate of the proximity function is

$$\hat{\phi}_g(\mathbf{y}_t) \quad = \quad \frac{1}{n_g}\sum_{j=1}^{n_g} d^2(\mathbf{y}_t,\mathbf{y}_{gj}) - \frac{1}{2n_g^2}\sum_{j,j'=1}^{n_g} d^2(\mathbf{y}_{gj},\mathbf{y}_{gj'}) \tag{24}$$

References

[1] B. Blankertz, 'BCI competition III results -Data set V-,' 2005 [Online]. Available: http://ida.first.fhg.de/projects/bci/competition_iii/results/

[2] J. D. Bayliss, 'Use of the evoked potential P3 component for control in a virtual apartment,' *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 11, pp. 113–116, 2003.

[3] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis, 'Learning to control a brain-machine interface for reaching and grasping by primates,' *PLoS Biol.*, vol. 1, pp. 193–208, 2003.

[4] J. del R. Millán, F. Renkens, J. Mouriño, and W. Gerstner, 'Noninvasive brain-actuated control of a mobile robot by human EEG,' *IEEE Trans. Biomed. Eng.*, vol. 51, pp. 1026–1033, 2004.

[5] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. Kübler, J. Perelmouter, E. Taub, and H. Flor, 'A spelling device for the paralysed,' *Nature*, vol. 398, pp. 297–298, 1999.

[6] J. del R. Millán, 'Adaptive brain interfaces,' *Comm. of the ACM*, vol. 46, pp. 75–80, 2003.

[7] B. Obermaier, G. R. Müller, and G. Pfurtscheller, 'Virtual keyboard controlled by spontaneous EEG activity', *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 11, pp. 422–426, 2003.

[8] B. Blankertz, K. R. Müller, D. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlögl, G. Pfurtscheller, J. del R. Millán, M. Schröder, and N. Birbaumer, 'The BCI competition III: Validating alternative approaches to actual BCI problems,' *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 14, pp. 153–159, 2006.

[9] C. R. Rao, *Advanced Statistical Methods in Biometric Research*, Wiley, New York, 1952.

[10] C. M. Cuadras, J. Fortiana, and F. Oliva, 'The proximity of an individual to a population with applications in discriminant analysis,' *Journal of Classification*, vol. 14, pp. 117–136, 1997.

[11] J. del R. Millán,'On the need for on-line learning in brain-computer interfaces.' *Proc. Int. Joint. Conf. on Neural Networks*, 2004.

[12] M. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*, Prentice-Hall, Upper Saddle River, NJ, 1993.

[13] V. Digalakis, J. R. Rohlicek and M. Ostendorf, 'ML Estimation of a Stochastic Linear System with the EM Algorithm and its Application to Speech Recognition', *IEEE Trans. Speech and Audio Proc.*, vol. 1, pp. 431–442 ,1993.

[14] C. M. Hajiyev and F. Caliskan, 'Fault detection in flight control systems based on the generalized variance of the Kalman filter innovation sequence,' *Proc. Amer. Cont. Conf.*, 1999.

[15] D. Verma and R. P. N. Rao, 'Goal-based imitation as probabilistic inference over graphical models,' *Advances in Neural Information Processing Systems 18 (NIPS 2005).*