HAL open science

FuRIA: An inverse Solution based Feature Extraction Algorithm using Fuzzy Set Theory for Brain-Computer Interfaces

Fabien Lotte, Anatole Lécuyer, Bruno Arnaldi

To cite this version:

Fabien Lotte, Anatole Lécuyer, Bruno Arnaldi. FuRIA: An inverse Solution based Feature Extraction Algorithm using Fuzzy Set Theory for Brain-Computer Interfaces. IEEE Transactions on Signal Processing, 2009, 57 (8), pp.3253-3263. inria-00368282

HAL Id: inria-00368282

https://inria.hal.science/inria-00368282v1

Submitted on 16 Mar 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

FuRIA: An Inverse Solution based Feature Extraction Algorithm using Fuzzy Set Theory for Brain-Computer Interfaces

Fabien LOTTE, Anatole LÉCUYER and Bruno ARNALDI

***Abstract*—This paper presents FuRIA, a trainable feature extraction algorithm for non-invasive Brain-Computer Interfaces (BCI). FuRIA is based on inverse solutions and on the new concepts of fuzzy Region Of Interest (ROI) and fuzzy frequency band. FuRIA can automatically identify the relevant ROI and frequency bands for the discrimination of mental states, even for multiclass BCI. Once identified, the activity in these ROI and frequency bands can be used as features for any classifier. The evaluations of FuRIA showed that the extracted features were interpretable and can lead to high classification accuracies.**

***Index Terms*—brain-computer interface (BCI), feature extraction, inverse solution, fuzzy sets, electroencephalography (EEG)**

EDICS Category: Biomedical Signal Processing (BIO)

I. Introduction

Brain-Computer Interfaces (BCI) are communication systems that enable a user to send commands to a computer only by means of brain activity [1]. Designing a BCI requires a researcher to measure the brain activity and to identify patterns in it. Brain activity measurements are generally achieved using ElectroEncephaloGraphy (EEG) which is portable, non invasive, cheap and offers a good time resolution [1]. Hence, identifying patterns in EEG signals is a major challenge in the design of efficient BCI. Two key points are involved in this identification, namely, feature extraction and classification [2]. Feature extraction aims at describing the EEG signals by some relevant values called features while classification aims at automatically assigning a class to these features.

A good feature extraction algorithm should capture the relevant information related to each targeted brain activity pattern (or mental state) while filtering away noise or any unrelated information. Moreover, EEG signals are known to be highly subject-specific, in terms of spatial or spectral contents for instance [1]. Consequently, an ideal feature extraction algorithm for BCI should be trainable in the sense that it should learn and use subject-specific features.

Inverse solutions are methods that attempt to reconstruct the activity in the brain volume by using only scalp measurements [3]. Hence, they may localize the neural sources of activity within the brain, thus recovering a physiologically relevant information. Recently, inverse solutions were shown to be promising feature extraction methods for EEG-based BCI [4].

In this paper we propose a trainable feature extraction algorithm for BCI which relies on inverse solutions. This algorithm is called FuRIA which stands for **Fu**zzy **R**egion of **I**nterest **A**ctivity. FuRIA aims at automatically identifying what are, for a given subject, the relevant Regions Of Interest (ROI) and their associated frequency bands for the discrimination of mental states. The activity in these ROI and associated frequency bands can be used as features for any classifier.

The paper is organized as follows: Section II details the concept of inverse solution and reviews its use for BCI design. Section III describes in details the FuRIA algorithm we propose. Section IV presents the results of the FuRIA evaluations. Finally, Section V discusses the results and concludes.

II. Background: inverse solutions and BCI

A. Inverse solutions

When using EEG, the signals $m(t)$ ($m \in \Re^{N_e,1}$ with N_e being the number of electrodes used) recorded at time t on the scalp can be modeled by a linear combination of brain dipole activity $c(t)$ ($c \in \Re^{3\times N_v,1}$ with N_v being the number of dipoles considered). This is called the *forward problem* [3], which can be modeled by:

$$m(t) = Kc(t) \tag{1}$$

where K is an $N_e \times (3 \times N_v)$ matrix called the *leadfield* matrix which represents the physical properties (conduction) of the head. This matrix is a head model in which each dipole is modeled by a volume element known as a *voxel*. Typical head models are composed of thousands of voxels [3]. The $c(t)$ vector holds the orientation and amplitude of each dipole, according to the three dimensions of the head model space. Inverse solutions aim at estimating the brain dipole activity $\hat{c}(t)$ by using only the scalp measurements $m(t)$ and the leadfield matrix (head model) K, which can be modeled by:

$$\hat{c}(t) = Tm(t) \tag{2}$$

where T is the generalized inverse of K. As $N_v >> N_e$, this problem has no unique solution and additional constraints must be added to solve it. Depending on the constraints used, different inverse solutions are obtained which leads to different T matrices [5]. There are two main kinds of inverse solutions:

Copyright (c) 2009 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

B. Arnaldi is with IRISA-INSA Rennes, campus universitaire de beaulieu, 35042 Rennes, France. e-mail: arnaldi@irisa.fr. A. Lécuyer is with INRIA Rennes, campus universitaire de beaulieu, 35042 Rennes, France. e-mail: anatole.lecuyer@irisa.fr. During this work, F. Lotte was with IRISA-INRIA-INSA Rennes, France. He now is with I2R, 1 Fusionopolis way, Connexis, 138632, Singapore. e-mail: fabien.lotte@gmail.com

distributed solutions and equivalent dipole solutions [5] [3]. Distributed solutions estimate the amplitudes and orientations of a large number of voxels distributed in all the cortex or in all the brain whereas equivalent dipole solutions estimate the position, amplitude and orientation of few sources (typically one or two), each one modeled by an equivalent dipole.

Congedo has shown that with any linear and discrete inverse solution the amplitude γ_v (also known as current density) in voxel v could be formulated as a quadratic form [6]:

$$\gamma_v(t) = m(t)^T Q_v m(t) \tag{3}$$

Here, Q_v is an $N_e \times N_e$ matrix denoted as the *inverse operator* for voxel v [6]. The superscript T denotes transpose. Typically $Q_v = T_v^T T_v$ with T_v being the vth row of T. Such a γ_v measure has been used as a feature for BCI (see next section). The current density in a given Region Of Interest (ROI) Ω, i.e., in a set of voxels, can be computed as follows:

$$\gamma_\Omega(t) = \sum_{v \in \Omega} \gamma_v(t) = m(t)^T Q_\Omega m(t) \text{ with } Q_\Omega = \sum_{v \in \Omega} Q_v \tag{4}$$

This notation is very convenient as it allows very fast computations, whatever the number of voxels in the ROI Ω. Actually the Q_Ω matrix can be computed offline, and the size of this matrix depends only on the number of electrodes used.

B. A review of inverse solution-based BCI

Concerning BCI, inverse solutions extract physiologically relevant information which appears as an attractive type of feature. Recently, a few studies have focused on inverse solutions for feature extraction and have obtained promising results [4] [7] [8]. In order to design BCI, inverse solutions are generally used in two different ways:

- As a direct feature extraction technique: in this case, either the brain current density computed in a number of ROI [9] or the neural source positions [10] [8] are used as features to identify a mental state.
- As a preprocessing method preceding feature extraction: in this case, the inverse solution is used to estimate $c(t)$ from which the features are extracted [11] [12] [7].

These methods have all obtained very satisfying results. Moreover, it has been observed that extracting features from $c(t)$ (the source domain) is more efficient than extracting them directly from $m(t)$ (the sensor domain) [11] [7] [13].

In spite of these promising results, some limitations remain. Indeed, current methods are either general-purpose, i.e., they have the ability to deal with any kind of mental state, or they generate a consise set of discriminative features but rarely both at the same time. Several methods require strong a priori knowledge on the neurophysiological mechanisms involved by the mental states used, and hence, are not general-purpose at all [10] [8] [12]. With these methods, the ROI to be used must be defined beforehand and by hand. These methods are currently limited to the use of mental states that involve the motor and sensorimotor areas of the brain. Moreover, a non-general-purpose method, exclusively based on a priori knowledge, will not be able to adapt to each subject's specificities, and will have, most probably, non-optimal performances.

A few general-purpose methods, based on distributed inverse solutions, have been proposed [11] [7] [13]. Their main drawback is that they must extract one or several features per voxel, which generates a very large number of features. Thus, this requires the use of feature selection techniques [11] [7] [13]. Even though this solution gives good results, the number of features used remains generally relatively high, particularly in comparison with the number of features extracted by non-general-purpose methods [10] [8]. Moreover, in these methods, all voxels are processed independently, whereas the current densities in neighboring voxels are generally correlated. As such, these voxels should be gathered in brain regions.

Congedo *et al* have proposed a method which is general-purpose and generates a small set of discriminative features, which are determined after gathering the voxels into a few non-overlapping ROI [9]. However, this method still needs improvements as it is not completely automatic and currently limited to the use of two ROI [9].

III. The FuRIA feature extraction algorithm

FuRIA is an inverse solution-based algorithm which can learn and use subject-specific features for mental state classification. A feature extracted with FuRIA corresponds to the activity in a given ROI and its associated frequency band. Contrary to existing methods, FuRIA can automatically identify relevant ROI, as well as the frequency bands in which these ROI current densities are discriminant. Finally, FuRIA also introduces the concepts of fuzzy ROI and fuzzy frequency bands to obtain increased classification performances.

FuRIA aims at being modular in the sense that various kinds of inverse solutions could be used within it. This section briefly describes the inverse solutions that could be used within FuRIA and the specific one that we used in our implementation. It then describes in details the FuRIA feature extraction algorithm.

A. Inverse solutions for FuRIA

Equivalent dipole solutions have been popularly employed for BCI as they can obtain satisfactory results with only two features and without using a trained classifier [10] [8]. However, FuRIA is based on linear and distributed inverse solutions. Actually, distributed solutions enable the use of a large number of dipoles rather than a few equivalent dipoles. As such they provide more information and are more flexible.

On the other hand, the use of linear inverse solutions appears as essential for BCI applications. Indeed, the computational demand of non-linear inverse solutions makes them inappropriate for online BCI operation. Several linear and distributed inverse solutions have been used for BCI, such as ELECTRA [11], sLORETA (standardized low resolution electromagnetic tomography) [9] or the depth-weighted minimum norm technique [12]. FuRIA can be used with any of them.

In our implementation of FuRIA, we used sLORETA which is an instantaneous, discrete and linear inverse solution [14]. sLORETA is known to have very good localization properties

[14] including no localization bias in the presence of measurement and biological noise [15]. Moreover, it has been proven experimentally that sLORETA is suitable for the design of EEG-based BCI [9]. With sLORETA, the *inverse operator* Q_v is computed by $Q_v = T_v^T S_v^{-1} T_v$ with $S_v = T_v K_v$, T being obtained with the minimum-norm method [14] [6]. More details on sLORETA can be found in [14] [15]. Even if we used sLORETA, it should be stressed that the principle of the FuRIA algorithm is not dependent on the inverse solution chosen. Consequently, any other distributed and linear inverse solution could be used instead of sLORETA.

B. Overview of the FuRIA algorithm

1) Training of FuRIA: To be used, FuRIA has first to be trained using a set $\Theta = \{(m(t), C)_{1..N_t}\}$ of labelled training data. C is the class of $m(t)$, i.e., the mental state of the subject while $m(t)$ was recorded. This training phase aims at finding subject-specific ROI Ω_l and frequency bands Φ_l that contain the most relevant information for mental state discrimination. This phase is accomplished offline, in three steps:

- **Identification of statistically discriminant voxels and frequencies:** The goal of this step is to identify the ordered pairs $w_k = (f_i, v_j)$ $(k \in [1, N_k])$, with f_i being a frequency and v_j being a voxel, with the largest discriminative power. In order to do so, we rely on a statistical test for comparing, between the different classes, the mean current density in each frequency f_i $(i \in [1, N_f])$ and in each voxel v_j $(j \in [1, N_v])$.
- **Creation of ROI and frequency bands:** This step aims at clustering the ordered pairs w_k selected in the previous step into a smaller number of ordered pairs W_l $(l \in [1, N_w])$. Here $W_l = (\Phi_l, \Omega_l)$, with Φ_l being a frequency band, i.e., a set of frequencies, and Ω_l being a ROI, i.e., a set of voxels. In order to do so, we rely on a clustering algorithm for finding clusters of voxels and frequencies and for transforming these clusters into ROI associated to frequency bands. Thus, by the end of this step, we have created a set of N_w ordered pairs W_l (one pair per cluster) in which each ROI is associated to a single frequency band. This frequency band should gather the frequencies in which the activity of the ROI voxels is discriminant.
- **Fuzzification of ROI and frequency bands:** The found ROI Ω_l are turned into fuzzy ROI $\tilde{\Omega}_l$ and the frequency bands Φ_l into fuzzy frequency bands $\tilde{\Phi}_l$. This aims at giving more importance to the more discriminant voxels and frequencies, while still using the information contained in the less discriminant ones. The overall objective is to increase the discriminative power of the W_l pairs.

2) Use of FuRIA for feature extraction: Once the fuzzy pairs $\tilde{W}_l = (\tilde{\Phi}_l, \tilde{\Omega}_l)$ have been identified, FuRIA can be used for feature extraction.

The lth feature extracted is the average current density in the fuzzy ROI $\tilde{\Omega}_l$ after band-pass filtering EEG signals in the associated fuzzy frequency band $\tilde{\Phi}_l$. All these steps are detailled hereafter.

In this paper, we evaluate a specific implementation of FuRIA. However, it is worth noting that other implementations could be used (e.g., different clustering algorithms or different statistical tests) as long as they are consistent with the algorithm proposed.

C. First training step: identification of statistically discriminant voxels and frequencies

1) Algorithm: The first training step of FuRIA aims at identifying the pairs w_k of voxels v_j and frequencies f_i which are the most discriminant, i.e., the pairs of voxels and frequencies whose current density is the most different between classes. To do so, we perform a statistical test that compares the mean current density between classes for each pair $w_k = (f_i, v_j)$. To this end, each training EEG record $m(t)$ passes through the following procedure (see Fig. 1):

1) $m(t)$ is decomposed into frequency bands by using a set of filters h_i. Each filter h_i is a 2-Hz wide band-pass filter centered on frequency f_i. We denote as $m_i(t)$ the signal resulting from the filtering of $m(t)$ by h_i.
2) the current density $\gamma_{i,j}(t)$ in voxel v_j, for frequency f_i is computed using the inverse solution:

$$\gamma_{i,j}(t) = m_i(t)^T Q_{v_j} m_i(t) \tag{5}$$

3) finally, $\gamma_{i,j}(t)$ is averaged over a time window of interest, which starts at sample t_0 and is N_s samples long. The obtained value is then log-transformed:

$$< \gamma_{i,j} >= \log\left(\frac{1}{N_s}\sum_{t=t_0}^{t_0+N_s} \gamma_{i,j}(t)\right) \tag{6}$$

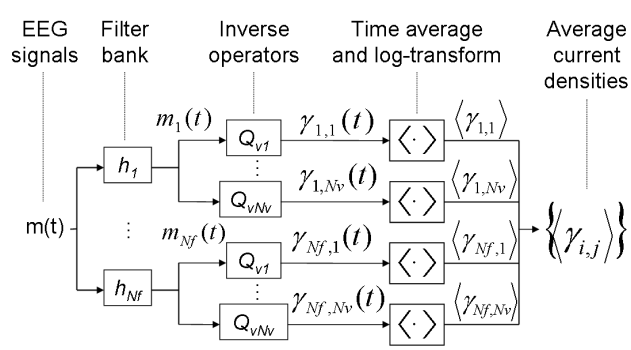

Fig. 1. Computation of the average current densities $< \gamma_{i,j} >$ in all frequencies f_i and voxels v_j, from a training data m(t).

All $< \gamma_{i,j} >$ are then gathered into statistical samples according to the label of their corresponding training record $m(t)$. These samples are compared using the statistical test. In other words, this statistical test compares the mean $< \gamma_{i,j} >$ between classes and thus gives the discriminative power of each pair w_k. Pairs w_k which obtained a p-value higher than a given (user-selected) threshold α are not considered anymore in the remaining of the training process. The other pairs are denoted as "significant". This procedure should remove numerous voxels and frequencies and should only keep the ones that are related to the mental states considered. It should

be noted that the w_k selected can be different from one subject to the other.

2) Implementation: In our implementation of FuRIA, the h_i filters were either Finite Impulse Response (FIR) or Infinite Impulse Response (IIR) filters. We used windowed sinc filters as FIR filters [16] and Yule-Walker filters as IIR filters [17]. These filters were chosen as they enable the design of custom filters, which is needed for the fuzzification step (see section III-E). Concerning the statistical test used, we employed Multiple Comparisons Randomization (MCR) tests as described by Holmes [18]. More precisely, for BCI with only 2 classes, we used MCR t-tests whereas for multiclass BCI (with 3 or more classes) we used MCR ANOVA (ANalysis Of VAriance), as ANOVA can compare multiple conditions, and as such, can deal with more than 2 classes.

D. Second training step: creation of ROI and frequency bands

1) Algorithm: This step aims at gathering pairs of significant voxels and frequencies into pairs of ROI Ω_l $(l \in [1, N_w])$ and frequency bands Φ_l, each ROI being associated to a single frequency band. Basically, a given ROI would gather significant voxels, and the frequency band associated to this ROI would gather the frequencies in which these voxels activity is discriminant. Creating such ROI and frequency bands aims at obtaining a compact feature set. Indeed, using the activity in a few ROI and frequency bands as features should lead to much fewer features than when considering voxels and frequencies alone, as generally done [11] [7]. Moreover, the activities in neighboring voxels and frequencies tend to be statistically correlated [5]. Thus, it may be appropriate to use these voxels and frequencies together rather than independently.

We would like to gather voxels belonging to the same neural source into the same ROI. We would also like to gather into a single frequency band the frequencies in which each ROI is discriminant. Thus, it is desirable to find clusters gathering both voxels and frequencies. To do so, we associate to each significant pair $w_k = (f_i, v_j)$ the feature vector $[x_j, y_j, z_j, f_i]$ in which x_j, y_j, z_j are the spatial coordinates of v_j. Each one of these 4 coordinates was normalized to zero mean and unit variance over all j in order to deal with the different ranges between space and frequencies. Then, we apply a given clustering algorithm to all these vectors. Finally, for each one of the N_w clusters obtained automatically, we gather into the same ROI Ω_l all the voxels whose spatial coordinates correspond to the x_j, y_j, z_j coordinates of one of the vectors from this cluster. We also associate to this ROI Ω_l the frequency band $\Phi_l = [f_{min}, f_{max}]$. Here, f_{min} and f_{max} are respectively the minimal and maximal value of the coordinate f_i among all the vectors belonging to this cluster. This clustering gives a set of ordered pairs $W_l = (\Phi_l, \Omega_l)$ which are expected to be discriminant.

2) Implementation: When using sLORETA, the neural sources tend to appear as local maximums of the current density [14]. Hence, we assume they would also appear as local maximums of the statistics obtained in output of the statistical test. Even though it is only an assumption, this seems to be often verified in practice (see results in Section IV-C for instance). We used Mean Shift as the clustering algorithm since it gathers vectors attracted by the same local maximum of the underlying density function [19]. However, as the voxels coordinates and frequencies considered are regularly spaced within their numerical domain, the underlying density function for the vectors $[x_j, y_j, z_j, f_i]$ will be relatively flat, and thus will prevent a proper use of Mean Shift. To cope with this problem, we used a slightly modified version of the Mean Shift algorithm for clustering. This slight modification simply consists in replacing $\hat{D}$, the standard density estimate at point P used in Mean Shift:

$$\hat{D}(P) = \frac{1}{N_k H^d} \sum_{k=1}^{N_k} \chi\left(\frac{1}{H}(P - P_k)\right) \tag{7}$$

by $\tilde{D}$, a weighted density estimate at point P :

$$\tilde{D}(P) = \frac{1}{N_k H^d} \sum_{k=1}^{N_k} s_k \cdot \chi\left(\frac{1}{H}(P - P_k)\right) \tag{8}$$

with H being the smoothing parameter, χ a kernel function, here the Epanechnikov kernel (which was found to be the optimal kernel for Mean Shift [19]), P the current vector, P_k, the kth vector from the data set, and d the dimensionality. Finally $s_k = 1 - p_k$, with p_k being the p-value obtained by the w_k pair, during the statistical test performed in the previous step. The vector P_k is the vector corresponding to the w_k pair. This leads to the following form for the *sample mean shift* vector :

$$M_H(P) = \frac{1}{n_P} \sum_{P_i \in S_H(P)} s_k \cdot (P_k - P) \tag{9}$$

Here, $S_H(P)$ is the d-dimensional sphere of radius H centered at P, with n_P vectors inside. This weighted version of Mean Shift will gather into the same cluster all the vectors attracted by the same local maximum of the statistics. As such, this version of Mean Shift aims at gathering altogether the vectors corresponding to the same neural source, as these sources are assumed to be local maximums of the statistics (see above).

E. Third training step: fuzzification of ROI and frequency bands

The last training step of FuRIA consists in fuzzifying the previously obtained ROI Ω_l and frequency bands Φ_l. Actually, a ROI can be seen as a conventional (or "crisp") sets of voxels whereas a frequency band can be seen as a crisp set of frequencies. However, in a ROI or in a frequency band, all the voxels and frequencies do not have the same discriminative power. Nevertheless, all these elements still carry information that could be used, making it hard to choose which of them should be kept. Consequently, we believe that all significant voxels and frequencies should be used, but the voxels and frequencies that are less discriminant should belong "less" to their ROI and frequency band than the others. Thus, we propose to consider ROI and frequency bands as fuzzy sets [20] of voxels and frequencies, in which all voxels and frequencies are given a degree of membership into the ROI or frequency band to which they belong. We

denote such kinds of ROI and frequency bands as fuzzy ROI and fuzzy frequency bands. Thus, a fuzzy membership function μ is associated to each ROI and each frequency band.

1) Design of fuzzy ROI and fuzzy frequency bands from a given fuzzy membership function:

a) Algorithm: As mentionned above, a crisp ROI Ω_l is a set of voxels which current density is computed according to Eq. 4. A fuzzy ROI $\tilde{\Omega}_l$ is not defined by a set of voxels anymore but by a fuzzy membership function μ_{Sl}. This function provides the degree of membership, in $[0,1]$, of any existing voxel to the fuzzy ROI $\tilde{\Omega}_l$. Contrary to crisp ROI for which the activity in all its voxels is used equally (see Eq. 4), we define the current density $\gamma_{\tilde{\Omega}_l}(t)$ in a fuzzy ROI as follows:

$$\gamma_{\tilde{\Omega}_l}(t) = \sum_{j=1}^{N_v} \mu_{Sl}(v_j)\gamma_{v_j}(t) \tag{10}$$

This leads to:

$$\gamma_{\tilde{\Omega}_l}(t) = m(t)^T Q_{\tilde{\Omega}_l} m(t) \text{ with } Q_{\tilde{\Omega}_l} = \sum_{j=1}^{N_v} \mu_{Sl}(v_j) Q_{v_j} \tag{11}$$

Similarly, we associate a fuzzy membership function μ_{Fl} to each frequency band Φ_l. The function μ_{Fl} provides the degree of membership, in $[0,1]$, of any existing frequency to the fuzzy frequency band Φ_l. We can note that this function has exactly the same form as the magnitude response of a digital filter. This means that to band-pass filter a signal in a given fuzzy frequency band we have to design a custom digital filter that has the desired fuzzy membership function μ_{Fl} as magnitude response.

b) Implementation: We used the window technique to automatically design FIR filters from the desired magnitude response, i.e., from a fuzzy membership function associated to a fuzzy frequency band [16]. To design the IIR filters, we used the Yule-Walker method [17]. This explains why we used a windowed-sinc FIR filter or a Yule-Walker IIR filter for the first training step of FuRIA (see section III-C).

2) Set up of the fuzzy membership functions:

a) Algorithm: To determine the kind of fuzzy membership functions to be used as well as their parameters, we first compute the discrimination scores d_{v_j} and d_{f_i} of each voxel v_j and frequency f_i respectively, for each pair $W_l = (\Phi_l, \Omega_l)$:

$$d_{v_j} = \frac{1}{N_f} \sum_{f_i \in \Phi_l} s_{v_j,f_i} \text{ and } d_{f_i} = \frac{1}{N_v} \sum_{v_j \in \Omega_l} s_{v_j,f_i} \tag{12}$$

where $s_{v_j,f_i} = 1 - p_{j,i}$ with $p_{j,i}$ the p-value obtained during the first training step, for voxel v_j at frequency f_i. In order to highly emphasize the contribution of the most discriminative voxels and frequencies, which are generally much less numerous than others, we chose exponential fuzzy membership functions that are Gaussian:

$$\mu_{S_l}(v_j) = \begin{cases} \exp(-\frac{1}{2}(\frac{d_{v_j} - d_{vmax}}{\sigma_v})^2) & v_j \in \Omega_l \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

$$\mu_{F_l}(f_i) = \begin{cases} \exp(-\frac{1}{2}(\frac{d_{f_i} - d_{fmax}}{\sigma_f})^2) & f_i \in \Phi_l \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

where d_{vmax} and d_{fmax} are the maximal scores among voxels and frequencies respectively. The values of the σ should be optimized in order to maximize the discriminative power of the pairs $\tilde{W}_l = (\tilde{\Phi}_l, \tilde{\Omega}_l)$. Consequently, we use $\sigma_v = \frac{1}{card(\Omega_l)}$ and $\sigma_f = \frac{1}{card(\Phi_l)}$ as initial values, and then optimize these values using the adaptive gradient ascent procedure described in Algorithm 1. Here λ_f and λ_v are positive learning rates,

Algorithm 1 Adaptive gradient ascent algorithm (ϵ_f,ϵ_v,λ_f,λ_v)

$newF \leftarrow F(\sigma_v, \sigma_f)$
repeat
  $oldF \leftarrow newF$
  $\Delta F_f \leftarrow F(\sigma_v, \sigma_f) - F(\sigma_v, \sigma_f + \epsilon_f)$
  $\sigma_f \leftarrow \sigma_f - \lambda_f \frac{\Delta F_f}{\epsilon_f}$ $\{\sigma_f$ update$\}$
  $newF \leftarrow F(\sigma_v, \sigma_f)$
  if $newF < oldF$ **then**
    $\lambda_f \leftarrow \frac{\lambda_f}{2}$ $\{\lambda_f$ Adaptation$\}$
  else
    $\lambda_f \leftarrow \lambda_f + 0.1\lambda_f$ $\{\lambda_f$ Adaptation$\}$
  end if
  $oldF \leftarrow newF$
  $\Delta F_v \leftarrow F(\sigma_v, \sigma_f) - F(\sigma_v + \epsilon_v, \sigma_f)$
  $\sigma_v \leftarrow \sigma_v - \lambda_v \frac{\Delta F_v}{\epsilon_v}$ $\{\sigma_v$ update$\}$
  $newF \leftarrow F(\sigma_v, \sigma_f)$
  if $newF < oldF$ **then**
    $\lambda_v \leftarrow \frac{\lambda_v}{2}$ $\{\lambda_v$ Adaptation$\}$
  else
    $\lambda_v \leftarrow \lambda_v + 0.1\lambda_v$ $\{\lambda_v$ Adaptation$\}$
  end if
until $|\frac{\Delta F_f}{\epsilon_f}| \leq 0.001$ and $|\frac{\Delta F_v}{\epsilon_v}| \leq 0.001$

and ϵ_v and ϵ_f are small positive increments used to estimate the derivatives of the function F. This function F is the fitness function that we want to maximize and that evaluates the discriminative power of a given pair $\tilde{W}_l$. This fitness function is equal to the statistics obtained with a statistical test that compares the current density in $\tilde{\Omega}_l$ and $\tilde{\Phi}_l$ between the different classes. More precisely, for each training record $m(t)$, this record is first band-pass filtered in the $\tilde{\Phi}_l$ frequency band by using the corresponding IIR or FIR filter. Then, the current density in $\tilde{\Omega}_l$ is computed using Eq. 10, and averaged over a given time window and log-transformed as in Eq. 6. One should note that the obtained values $< \gamma_{\tilde{\Omega}_l,\tilde{\Phi}_l} >$ depend on σ_v and σ_f which are used to compute the band-pass filter and the ROI current density. These $< \gamma_{\tilde{\Omega}_l,\tilde{\Phi}_l} >$ are then arranged by class label. The statistical test finally compares the mean value of these $< \gamma_{\tilde{\Omega}_l,\tilde{\Phi}_l} >$ between the different classes, the null hypothesis H_0 being "the mean value of the $< \gamma_{\tilde{\Omega}_l,\tilde{\Phi}_l} >$ is not different between the classes". The obtained statistics is used as the value of the fitness function F. Thus, algorithm 1 selects the values of σ_f and σ_v that maximize the discriminative power of a given $\tilde{W}_l$. Naturally, this procedure is performed for each pair W_l. A gradient ascent optimization seemed appropriate here as we experimentally observed that

the fitness function F was not monotonic. Rather, this fitness function generally had an optimum for small values of the σ parameter.

b) Implementation: Concerning the statistical test used to compute the fitness function F, we used a t-test for binary BCI and an ANOVA for multiclass BCI. These tests are the same ones as those used during the first training step (see III-C). Concerning the values of $\epsilon_f, \epsilon_v, \lambda_f$ and λ_v, we performed extensive experimental tests and found that $\epsilon_f = \epsilon_v = 0.0001$ and $\lambda_f = \lambda_v = 10^{-5}$ were appropriate values. We used these values in all our experiments.

At the end of this offline training, a set of fuzzy ROI $\tilde{\Omega}_l$ associated to fuzzy frequency bands $\tilde{\Phi}_l$ has been identified. They can now be used for feature extraction, possibly online.

F. Feature Extraction with FuRIA

Once the training is achieved, feature extraction with FuRIA consists in computing the current density in each fuzzy ROI and fuzzy frequency band and in using these current density values as features. More formally, it consists in filtering the EEG signals $m(t)$, once for each one of the N_w fuzzy ROI $\tilde{\Omega}_l$ obtained, using the FIR or IIR filter corresponding to $\tilde{\Phi}_l$. Then, $< \gamma_{\tilde{\Omega}_l,\tilde{\Phi}_l} >$, the current density in $\tilde{\Omega}_l$, is computed using Eq. 10 and averaged over a given time window and log-transformed as in Eq. 6. The N_w current densities $< \gamma_{\tilde{\Omega}_l,\tilde{\Phi}_l} >$ are then concatenated into a N_w dimensional feature vector $[< \gamma_{\tilde{\Omega}_1,\tilde{\Phi}_1} >, < \gamma_{\tilde{\Omega}_2,\tilde{\Phi}_2} >, \ldots, < \gamma_{\tilde{\Omega}_{N_w},\tilde{\Phi}_{N_w}} >]$. Such a feature vector can then be used as an input of any classifier, e.g., a Support Vector Machine, this classifier being in charge of estimating the class of $m(t)$ from the features. Figure 2 summarizes the principle of feature extraction using FuRIA.

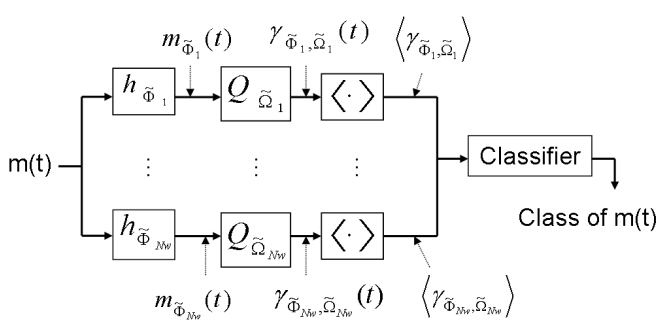

Fig. 2. Feature extraction with FuRIA.

G. Model selection

In our implementation of FuRIA, two hyperparameters should be defined by the user: 1) α, the significance threshold used in the statistical test of the first training step (see Section III-C) and 2) H, the smoothing parameters used in the Mean Shift clustering during the second training step (see Section III-D). The values of these hyperparameters have an impact on the number of features N_w and on the extension and shape of the ROI and frequency bands. As this impacts the performance of the recognition algorithm, we must select the most appropriate hyperparameters. A solution could be to test several values for H and α and select the couple that enables the best classification on a training set, estimated using a given classifier and Cross Validation (CV). However, we noticed that this method favors models with numerous features, which is not desirable. Furthermore, we observed experimentally that there were generally models with a classification accuracy only slightly lower than the best one, but with much less features. Indeed, if we plot the number of features versus the classification accuracy for several models, the resulting curve tends to be relatively flat for large numbers of features, and suddendly decreases for a smaller number of features (see Figure 3). Ideally, we would like to use the model

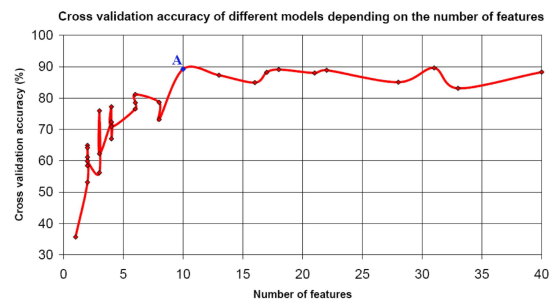

Fig. 3. Example of a plot of the number of features versus the 10*10 fold cross validation accuracy of several FuRIA models. The point A corresponds to the model we would like to automatically select. These data come from subject S1 from the BCI competition 2005 (see Section IV-A2).

corresponding to the point of the curve situated just before this sudden decrease of classification accuracy (point A, in blue, on Figure 3), as it would be the best tradeoff between a high classification accuracy and a small number of features. Indeed, we believe that such a model with few features should be prefered as it is probably more robust, less computationally demanding, more easily interpretable and it should ease the training of the classifier. Consequently, we propose a simple model selection criterion C_p:

$$C_p = 2 \cdot A - N_w \tag{15}$$

where A is the accuracy (in percent) obtained using CV on a training set. The model with the highest C_p is the one that should be prefered. Thus, this criterion is still based on CV but it penalizes models with many features. We also consider that models with a number of features lower than the number of mental states should be avoided. Actually, we consider that a mental state is generated by at least one brain region. In pratice, we observed that this proposed criterion gave satisfactory results. This is shown in the next section which is devoted to the evaluation of FuRIA.

IV. Evaluations of FuRIA

These evaluations have two objectives. First we want to assess the impact of the different hyper-parameters as well as the contribution of the fuzzification processes on the performances. Then we also want to globally assess the efficiency of FuRIA, by comparing a BCI based on FuRIA with other state-of-the-art BCI systems used during BCI competitions.

In order to assess FuRIA, we evaluated it on four different subjects, available from two data sets of BCI competitions 2003 and 2005. For all these evaluations, we worked with a standard head model, composed of three concentric spheres and containing 2394 voxels. This head model has been obtained using LORETA-Key, a software dedicated to inverse solutions (see [21] for more details on this software). It could have been better to use subject-specific head models, however, the information needed to build such models was not available in the BCI competition data. For the training of FuRIA, we considered the frequencies located in the 3-45 Hz frequency band with a step of 1 Hz between two consecutive frequencies. Concerning the MCR tests, we used a value of 1000 for the number of random permutations. The FuRIA features were classified using a Gaussian Support Vector Machine (SVM), as SVM is one of the most popular and efficient classifiers used for BCI [2]. When dealing with multiclass problems, several SVM were combined using the One-Versus-the-Rest (OVR) scheme in order to design a multiclass classifier. The optimal SVM hyperparameters were selected using 10*10 fold stratified cross validation. The descriptions of the data sets and the results obtained are presented in the following sections.

A. EEG data sets

1) BCI competition 2003 - data set IV: The first data set used was the EEG data set IV of the BCI competition 2003 [22], provided by the Berlin group. These data contain EEG signals recorded while a subject was performing self-paced left and right finger tapping tasks. EEG signals were sampled at 100 Hz, recorded using 28 electrodes and comprised the 500 ms before the actual movement. 314 trials were available for training and 100 for testing. The goal of the competitors was to forecast, for each trial, the hand that was used. For this data set, we used FuRIA to learn and extract features on the last 250 ms time window of the data, i.e., we used $t_0 = 25$ and $N_s = 25$ in Eq. 6. According to several studies, this time window should be the most informative [23] [9]. Here, we used FIR and IIR filters with 24 points and an order 8 respectively.

2) BCI competition 2005 - data set IIIa: The second data set used was the EEG data set IIIa of the BCI competition 2005 [24], provided by the Graz group [25]. These data were recorded while three subjects S1, S2 and S3, were performing a 4-class motor imagery task. They were instructed to imagine left hand, right hand, foot or tongue movements. For both training and testing, 60 trials were available per class. Trials were sampled a 256 Hz and were recorded using 60 electrodes. Each trial lasted 7 seconds without taking into account the inter trial periods of random lengths. The subjects were instructed to perform the motor imagery tasks during the last 3 seconds of each trial. For all subjects, we subsampled the data at 128 Hz, and used as time window for FuRIA these last 3 seconds, i.e., we used $t_0 = 512$ and $N_s = 384$ in Eq. 6. For training FuRIA, we ignored all trials contamined by artifacts, leaving approximatly between 25 and 45 trials per class, depending on the subject. Here, we used FIR and IIR filters with 50 points and an order 10 respectively.

B. Evaluation of the influence of hyperparameters and fuzzification processes

In this section, we evaluated FuRIA for different values of the hyperparameter α (see Section III-C), among $\{0.01, 0.05, 0.1, 0.25, 0.5\}$, and different values of the hyperparameter H (see Section III-D), among $\{0.75, 1, 1.25, 1.5, 1.75, 2\}$. The goal was to assess the impact of these hyperparameters on the classification performances and on the number of generated features. For all these data sets we also compared the results obtained when using FuRIA without the fuzzification process (i.e., using only crisp ROI and frequency bands), with only the spatial fuzzification, with only the frequential fuzzification and with the full (spatial and frequential) fuzzification. In the following, these four conditions are denoted as "Raw", "Freq", "Space" and "All" respectively. We also computed the results for both FIR and IIR filters. However, due to space limitations, only the results for the filters that gave the best results are presented.

1) BCI competition 2003 - data set IV: Tables I and II display the mean classification accuracies obtained on the test set when using FIR filters, for different values of α and H respectively. The mean number of features, denoted as "#Features", is also displayed. In each table, the best result for each condition and each subject is displayed in bold figures. The differences between the Raw condition and the fuzzy conditions, as revealed by a paired t-test, showed that all the fuzzy conditions performed better than the Raw condition on average. However, this difference is only significant for the "All" ($p < 0.05$) and "Space" ($p << 0.01$) conditions.

TABLE I
DATA SET IV, BCI COMPETITION 2003, TEST SET: CLASSIFICATION ACCURACY (%) AND NUMBER OF FEATURES FOR DIFFERENT VALUES OF α, AVERAGED OVER THE DIFFERENT VALUES OF H.

α	#Features	Raw	Freq	Space	All
0.01	10.17	**80.5**	**81**	82	80.33
0.05	10.17	79.5	80.67	82.67	**82.5**
0.1	9.17	77	78.17	**84.67**	81.83
0.25	9.67	76.67	77.17	83.17	76.17
0.5	8.83	72.17	74.83	78	77.33

TABLE II
DATA SET IV, BCI COMPETITION 2003, TEST SET: CLASSIFICATION ACCURACY (%) AND NUMBER OF FEATURES FOR DIFFERENT VALUES OF H, AVERAGED OVER THE DIFFERENT VALUES OF α.

H	#Features	Raw	Freq	Space	All
0.75	31.8	**83.6**	**82**	**83.4**	**82.6**
1	12.6	80.2	79.6	82.6	80.8
1.25	6.4	78.2	79	82.6	79.2
1.5	2.8	74.4	76.8	80	78.4
1.75	2	73.2	76.6	82	78.2
2	2	73.4	76.2	82	78.6

2) BCI competition 2005 - data set IIIa: Tables III and IV display the mean classification accuracies obtained for each of the three subjects on the test set, for different values of α and H respectively. Please note that, for subject S2, no results are presented for $\alpha = 0.01$ as no pair w_k was found significant with this threshold. Here again, only the results obtained with

the best filter are displayed, that is IIR filters for S1 and S2 and FIR filters for S3.

TABLE III
DATA SET IIIA, BCI COMPETITION 2005, TEST SET: CLASSIFICATION ACCURACY (%) AND NUMBER OF FEATURES FOR DIFFERENT VALUES OF α, AVERAGED OVER THE DIFFERENT VALUES OF H.

Subject	α	#Features	Raw	Freq	Space	All
S1	0.01	9.17	75.28	76.02	74.26	73.15
	0.05	9.33	**77.96**	**79.63**	76.76	**78.46**
	0.1	10.67	75.93	77.22	**79.17**	77.78
	0.25	7.83	60.93	67.96	72.31	68.98
	0.5	9	62.13	66.76	75.19	71.39
S2	0.05	7.83	**54.58**	44.31	**58.19**	55.278
	0.1	11.17	53.33	**50.97**	57.78	**56.11**
	0.25	11.67	50.97	43.89	47.22	45.69
	0.5	14.67	50.83	47.92	50.69	51.39
S3	0.01	6.17	71.67	69.72	71.94	68.75
	0.05	9.83	**73.06**	**76.94**	70.42	**77.78**
	0.1	9.17	70.42	72.08	**72.92**	72.5
	0.25	11.83	66.39	75	67.92	68.75
	0.5	12.5	64.17	71.39	68.47	67.08

TABLE IV
DATA SET IIIA, BCI COMPETITION 2005, TEST SET: CLASSIFICATION ACCURACY (%) AND NUMBER OF FEATURES FOR DIFFERENT VALUES OF H, AVERAGED OVER THE DIFFERENT VALUES OF α.

Subject	H	#Features	Raw	Freq	Space	All
S1	0.75	25.4	**86.22**	**88.67**	**87.44**	**89.33**
	1	12.2	84.22	85.67	86.56	85.89
	1.25	7.2	77.44	79.33	79.11	77.66
	1.5	5	70.89	73.67	76.11	72.66
	1.75	3.2	58.56	62.33	68.11	63.56
	2	2.2	45.33	51.44	55.89	54.56
S2	0.75	23.6	**61.39**	49.44	**60.83**	**56.94**
	1	13	58.96	51.46	58.33	56.25
	1.25	8.6	57.50	**55.28**	56.67	55.83
	1.5	5.2	54.44	53.06	55.56	55
	1.75	2.6	55.56	53.89	56.67	55
	2	1.4	56.11	45.83	54.44	50
S3	0.75	26	**81.67**	**80.17**	**83.5**	**81**
	1	13.4	78.33	80	80	80.17
	1.25	7.6	76	78.5	78.17	79.5
	1.5	5.6	74.33	78.5	77.83	79.83
	1.75	4	55.5	63.83	51.83	58.33
	2	2.8	49	57.17	50.67	47

3) Discussion: We performed a paired t-test to investigate the overall statistical differences between all the conditions, across all subjects, filter kinds and hyperparameter values. This revealed that globally, all fuzzy conditions performed better than the "Raw" one, and that this difference was significant ($p < 0.001$). This suggests that for a given set of ROI and frequency bands, fuzzifying them is likely to increase their classification performances. Both the "Space" and "All" conditions performed significantly better than the "Freq" one ($p < 0.05$). There was no significant difference between these two conditions ($p > 0.05$), even though the "Space" condition gave slightly better results than the "All" one, on average. We then suggest to use, by default, the spatial fuzzification only, or to try the different fuzzifications and select the most appropriate one for a given subject. The superiority of the spatial fuzzification over the frequential fuzzification could be due to the spatial resolution (thousands of voxels) being higher than the frequential resolution (tens of frequencies). As such, the spatial fuzzification could be more deeply exploited.

Concerning the effects of the threshold α, it can be noticed that the "Raw" condition reached its best performances using the traditional values $\alpha = 0.01$ or $\alpha = 0.05$ whereas the fuzzy conditions generally reached their best results for the values $\alpha = 0.05$ or $\alpha = 0.1$. This suggests that the fuzzification process enables to use efficiently the information contained in less discriminant voxels and frequencies in order to improve the performances. Regarding the results for the hyperparameter H, the best accuracy is almost always obtained for the lowest value of H. However, accuracies very close to these ones can be obtained with higher value of H and hence, much less features. This supports the use of the previously proposed model selection criterion which penalizes models with a large number of features (see section III-G).

C. Comparison with BCI competition results

In this section, we assess the global efficiency of the FuRIA features by comparing a BCI based on FuRIA features with BCI used by the BCI competition participants. To perform this comparison, we needed to select some parameters without the knowledge of the test set. Consequently, we only relied on cross validation scores on the available training sets. Thus, we selected the kind of filter (FIR or IIR) and the kind of fuzzification (spatial, frequential or both) according to their average cross validation score on the training set. To select the optimal hyperparameters H and α, we relied on the model selection criterion proposed in section III-G, Equation 15.

1) BCI competition 2003 - data set IV: Based only on the training set, the parameter selection procedure described above found that $H = 1.75$, $\alpha = 0.05$, FIR filters and only the spatial fuzzification was the most appropriate configuration. This resulted in only 2 features. These parameters led to an accuracy of 84% on the test set, that is, exactly the same score as the winner of the competition [23]. This suggests that the method is efficient, especially when considering the fact that only two features were used.

Fig. 4 displays the two ROI and frequency bands learnt by FuRIA. Interestingly, these ROI lie in the left and right motor areas, and the frequency bands lie in the β ($\simeq$ 13-30 Hz) band, which is consistent with the literature on the subject [23] [9] [26]. This suggests that the FuRIA features are interpretable features, which can be used to check what has been learnt or even to extract knowledge about the brain dynamics.

2) BCI competition 2005 - data set IIIa: Table V sums up the parameters used for each subject and selected using only the training set. The resulting number of features is also displayed. Interestingly, different parameters are needed for each subject. This highlights the high subject-specificity of EEG signals and the need to use trainable feature extraction methods. Concerning this data set, the goal of the participants was to provide a continuous classification, i.e., a class label for each time point, here from second 4 to second 7 of each trial. It was indeed during this period that the motor imagery tasks were performed. Hence, the FuRIA features were first learnt on the time window from second 4 to second 7 of each

Fig. 4. The fuzzy ROI (in red) and their corresponding frequency bands that were automatically obtained by using FuRIA for data set IV of BCI competition 2003. The brighter the red color of the voxel, the higher the voxel degree of membership $\mu_{Sl}(v_j)$. The brain is seen from the top, nose up. These pictures were obtained with the LORETA-Key software [21].

TABLE V
PARAMETERS USED FOR DATA SET IIIA FROM BCI COMPETITION 2005

	H	α	filter kind	fuzzification	feature number
S1	1	0.5	IIR	fuzzy space	11
S2	1	0.05	IIR	fuzzy space	12
S3	1.25	0.1	FIR	all fuzzy	5

trial. We classified each point by using the FuRIA features extracted from the last 1 second window. Classification outputs were also aggregated across time, which is known to improve accuracy [2]. In order to do so we used a different multiclass Gaussian SVM (made of several binary SVM combined using the OVR scheme) for each time point of the trial period from second 4 to second 7. These SVM were trained on the FuRIA features extracted on the 1 second window preceding their corresponding time point. Then, to classify each trial, the final output $O_f(t)$ at time t was estimated by using the individual outputs $O_i(t)$ of the SVM corresponding to the previous time points situated from second 4 to the present time point:

$$O_f(t) = \sum_{k=0}^{t} w_k O_i(k) \tag{16}$$

where the outputs O_i are vectors containing the output of each SVM used in the OVR scheme, i.e., one output per class. Here, $t = 0$ corresponds to second 4 of the trial. We defined $w_k = A_k - 25$ with A_k being the 10*10 fold CV accuracy (in percent) of the kth SVM on the training set. Hence, this method is a weighted combination of classification outputs across time. These weights were chosen so as to ignore contributions from randomly performing classifiers (with a CV error of 25 $\%$, as there are 4 classes) and emphasize contributions of well performing classifiers. The final class C_f attributed to a given point t was the one for which $C_f = argmax(O_f(t))$.

The performance measure used was the maximal classification accuracy. Performances obtained using our methods as well as performances obtained by BCI competition 2005 participants on data set IIIa [27] are reported for comparisons in table VI. All these participants also used SVM as classifiers (combined with other classifiers for the $2nd$ participant). For feature extraction, they all used a frequential information (band-pass filters or amplitude spectra) combined with spatial filters such as common spatial patterns, independant component analysis, and/or surface Laplacian [24]. As shown by table

TABLE VI
MAXIMAL CLASSIFICATION ACCURACY FOR THE TEST SETS OF DATA SET IIIA FROM BCI COMPETITION 2005 (%)

	S1	S2	S3	Mean
Winner	86.67	**81.67**	85.00	**84.44**
2nd	92.78	57.50	78.33	76.20
3rd	**96.11**	55.83	64.17	72.04
FuRIA	90.56	69.17	**88.33**	82.68

VI, our method outperformed the one of the overall winner of the competition on 2 subjects out of 3 (subjects 1 and 3) and even reached the best score among all participants on subject 3. However, the overall winner reached a really impressive score on subject 2 (the "worst" subject according to the general performances), leading him to the best overall results. Globally, our method reached the second position, with a score only slightly lower than the one of the winner. This shows the efficiency of FuRIA, especially when considering the few features used.

Fig. 5 displays the fuzzy ROI and corresponding fuzzy frequency bands automatically learnt by FuRIA for subject 3. Interestingly, the fuzzy ROI identified as relevant are located in the left and right motor areas, for frequency bands located within the μ (9-12 Hz) and β (13-30 Hz) rhythms. This is consistent with the literature on motor imagery [26] which, again, enhances the interpretability of the extracted features.

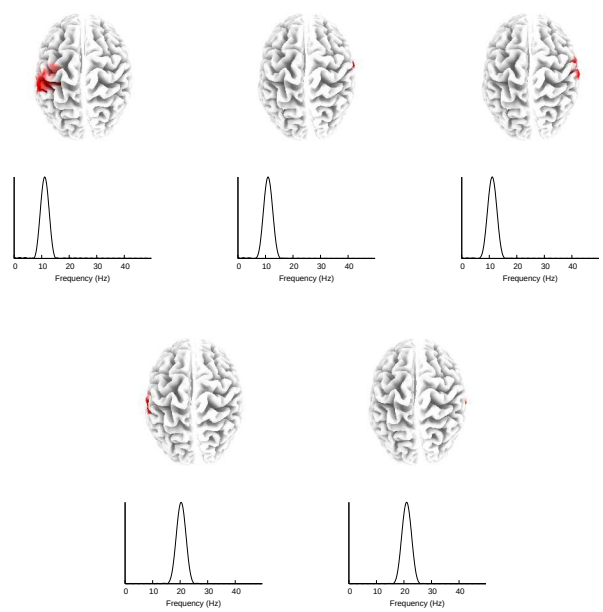

Fig. 5. The fuzzy ROI (in red) and their corresponding fuzzy frequency band (which is equivalent to the filter magnitude response) that were automatically obtained by using FuRIA, on subject 3 from data set IIIa of BCI competition 2005. The brighter the voxel red color, the higher the voxel degree of membership $\mu_{Sl}(v_j)$. The brain is seen from the top, nose up. These pictures were obtained with the LORETA-Key software [21].

V. Conclusion

This paper has presented FuRIA, a trainable feature extraction algorithm for Brain-Computer Interfaces which is based on inverse solutions. This algorithm can be trained to automatically identify relevant regions of interest and their associated frequency bands for the discrimination of mental states, in binary as well as in multiclass BCI. This paper also introduced the concepts of fuzzy ROI and fuzzy frequency bands that enable to use efficiently the available information and, thus, to increase the classification performances.

The evaluation of the proposed method, using sLORETA as the inverse solution and an SVM as classifier, showed its efficiency in terms of classification accuracy. Actually, the obtained results were comparable with those of BCI competition winners. Indeed, it seems that the inverse solution combined with the FuRIA training, acts as a spatial filter that removes the background activity and the noise not correlated with the targeted mental states. As such it focuses on relevant, subject-specific, brain activity features. An additionnal advantage of FuRIA is the interpretability of the learnt and extracted features. The main drawback of FuRIA is its long training process. Indeed, the discriminative power of several voxels and frequencies should be investigated, which can be long if the number of voxels, electrodes or training data is large, and if time consuming statistical tests such as the ones based on permutation tests are used. However, as this training is performed offline, this point does not seem critical.

In future works, it would be interesting to study the influence of both the spatial resolution (number of electrodes used, number of voxels in the chosen head model) and the frequential resolution (number of frequencies investigated) on the performances, in order to possibly reduce the training times and/or improve the quality of the feature extraction. It would also be interesting to take into account the temporal information in FuRIA. To this end, an attractive possibility would be to replace the classical band-pass filters by wavelets, which have proven to be particularly adpated for neuroelectric signals and especially EEG [28]. Finally, it could be interesting to use the dipole orientation in inverse solution-based BCI as current methods only consider the dipole amplitude.

Acknowledgements

This work was supported by the French National Research Agency within the Open-ViBE project and grant ANR05RNTL01601. The authors also thank Dr. Schlögl, Pr. Yanez-Suarez, Dr. Maby, Dr. Congedo, Dr. Gribonval, Ms. Rosendale and anonymous reviewers for their help and support.

References

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

[2] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for eeg-based brain-computer interfaces," *Journal of Neural Engineering*, vol. 4, pp. R1–R13, 2007.

[3] S. Baillet, J. Mosher, and R. Leahy, "Electromagnetic brain mapping," *IEEE Signal Processing Magazine*, vol. 18, no. 6, pp. 14–30, 2001.

[4] F. Lotte, A. Lécuyer, and B. Arnaldi, "Furia: A novel feature extraction algorithm for brain-computer interfaces using inverse models and fuzzy region of interest," in *Proc. of the 3rd IEEE-EMBS international Conference on Neural Engineering*, 2007, pp. 175–178.

[5] C. Michel, M. Murray, G. Lantz, S. Gonzalez, L. Spinelli, and R. G. de Peralta, "Eeg source imaging," *Clin Neurophysiol.*, vol. 115, no. 10, pp. 2195–2222, 2004.

[6] M. Congedo, "Subspace projection filters for real-time brain electromagnetic imaging," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 8, pp. 1624–1634, 2006.

[7] Q. Noirhomme, R. Kitney, and B. Macq, "Single trial eeg source reconstruction for brain-computer interface," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 5, pp. 1592–1601, 2008.

[8] B. Kamousi, Z. Liu, and B. He, "Classification of motor imagery tasks for brain-computer interface applications by means of two equivalent dipoles analysis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 13, pp. 166–171, 2005.

[9] M. Congedo, F. Lotte, and A. Lécuyer, "Classsification of movement intention by spatially filtered electromagnetic inverse solutions," *Physics in Medicine and Biology*, vol. 51, no. 8, pp. 1971–1989, 2006.

[10] L. Qin, L. Ding, and B. He, "Motor imagery classification by means of source analysis for brain computer interface applications," *Journal of Neural Engineering*, vol. 1, no. 3, pp. 135–141, 2004.

[11] R. Grave de Peralta Menendez, S. Gonzalez Andino, L. Perez, P. Ferrez, and J. Millán, "Non-invasive estimation of local field potentials for neuroprosthesis control," *Cognitive Processing*, vol. 6, pp. 59–64, 2005.

[12] F. Babiloni, F. Cincotti, M. Marciani, S. Salinari, L. Astolfi, A. Tocci, F. Aloise, F. D. V. Fallani, S. Bufalari, and D. Mattia, "The estimation of cortical activity for brain-computer interface: Applications in a domotic context," *Computational Intelligence and Neuroscience*, vol. 2007, 2007.

[13] M. Besserve, J. Martinerie, and L. Garnero, "Non-invasive classification of cortical activities for brain-computer interfaces: a variable selection approach," in *IEEE-ISBI' 2008*, 2008, pp. 1063–1066.

[14] R. Pascual-Marqui, "Standardized low resolution brain electromagnetic tomography (sloreta): technical details," *Methods and Findings in Experimental and Clinical Pharmacology*, vol. 24D, pp. 5–12, 2002.

[15] ——, "Discrete, 3d distributed, linear imaging methods of electric neuronal activity. part 1: exact, zero error localization," arXiv:0710.3341v2, Tech. Rep., 2007.

[16] S. W. Smith, *The Scientist & Engineer's Guide to Digital Signal Processing*. ISBN 0-9660176-3-3, 1997.

[17] B. Friedlander, , and B. Porat, "The modified yule-walker method of arma spectral estimation," *IEEE Transactions on Aerospace Electronic Systems*, vol. 20, no. 2, pp. 158–173, 1984.

[18] A. P. Holmes, R. C. Blair, J. D. Watson, and I. Ford, "Nonparametric analysis of statistic images from functional mapping experiments," *J. Cereb. Blood Flow Metab.*, vol. 16, pp. 7–22, 1996.

[19] D. Comaniciu and P. Meer, "Mean shift - a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[20] L. A. Zadeh, "Fuzzy sets," *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A. Zadeh*, pp. 19–34, 1996.

[21] R. Pascual-Marqui, "Loreta/sloreta website." [Online]. Available: http://www.unizh.ch/keyinst/NewLORETA/LORETA01.htm

[22] B. Blankertz, K. R. Müller, G. Curio, T. M. Vaughan, G. Schalk, J. R. Wolpaw, A. Schlögl, C. Neuper, G. Pfurtscheller, T. Hinterberger, M. Schröder, and N. Birbaumer, "The bci competition 2003: Progress and perspectives in detection and discrimination of eeg single trials," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1044–1051, 2004.

[23] Y. Wang, Z. Zhang, Y. Li, X. Gao, S. Gao, and F. Yang, "Bci competition 2003–data set iv: an algorithm based on cssd and fda for classifying single-trial eeg," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1081–1086, 2004.

[24] B. Blankertz, K. R. Muller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlogl, G. Pfurtscheller, J. D. R. Millan, M. Schroder, and N. Birbaumer, "The bci competition iii: Validating alternative approaches to actual bci problems," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 2, pp. 153–159, 2006.

[25] A. Schlögl, F. Lee, H. Bischof, and G. Pfurtscheller, "Characterization of four-class motor imagery eeg data for the bci-competition 2005," *Journal of Neural Engineering*, pp. L14–L22, 2005.

[26] G. Pfurtscheller and F. H. L. da Silva, "Event-related eeg/meg synchronization and desynchronization: basic principles," *Clinical Neurophysiology*, vol. 110, no. 11, pp. 1842–1857, 1999.

[27] A. Schlögl, "Results of the bci-competition 2005 for datasets iiia and iiib," Institute of Human-Computer Interface Graz University of

Technology, Tech. Rep., 2005. [Online]. Available: http://www.dpmi.tugraz.at/~schloegl/publications/TR_BCI2005_III.pdf

[28] V. Samar, A. Bopardikar, R. Rao, and K. Swartz, "Wavelet analysis of neuroelectric waveforms: A conceptual tutorial," *Brain and Language*, vol. 66, no. 1, pp. 7–60, 1999.

Fabien LOTTE Fabien Lotte obtained a M.Sc. degree and a PhD degree in computer sciences, both from the National Institute of Applied Sciences (INSA) Rennes, France, in 2005 and 2008 respectively. As a PhD candidate he was part of the BUNRAKU team at the French National Research Institute for Computer Science and Control (INRIA) and member of the OpenViBE project (www.irisa.fr/bunraku/OpenViBE/wiki) dedicated to brain-computer interfaces and virtual reality. In october and november 2008, he was a visiting PhD student at the Hirose-Tanikawa laboratory, the University of Tokyo, Japan, still working on brain-computer interfaces. Currently, he works as a research fellow in Singapore, at the Institute for Infocomm Research (I2R), Signal processing department, in the Brain-Computer Interface laboratory. His research interests include signal processing, machine learning, brain-computer interfaces and virtual reality.

Anatole LÉCUYER Anatole Lécuyer is a senior researcher at INRIA, the French National Institute for Research in Computer Science and Control (www.inria.fr), in Rennes, France. His main research interests include: virtual reality, 3D interaction, haptic feedback, pseudo-haptic feedback and brain-computer interfaces. In 1996, he graduated in France from Ecole Centrale Lille. He received his Ph.D. in Computer Science in 2001 from University of Paris XI, France. He spent one year as a research engineer in the French Commission for Atomic Energy, Fontenay-aux-Roses, France. At INRIA, he is a member of the BUNRAKU research team. He is currently the coordinator of the Open-ViBE National project on Brain-Computer Interfaces (www.irisa.fr/bunraku/OpenViBE/), and the leader of the Working Group on Haptic Interaction of INTUITION IST European Network of Excellence (www.intuition-eunetwork.net/).

Bruno ARNALDI Bruno Arnaldi received the Ph.D degree in Computer Science from the University of Rennes in 1988. For his thesis, he introduced physically based models in computer animation. In 1988 he became a full time INRIA researcher in computer graphics at IRISA/INRIA. From 1994 he became Professor of Computer Science at INSA de Rennes (Engineer School). He was head of Project SIAMES (Image Synthesis, Animation, Modelling and Simulation) and was chairman of the French association for Computer Graphics (AFIG - 140 members). He was also chairman of PERF-RV, the French National Platform dedicated to Virtual Reality. He is currently president of the French Association for Virtual Reality (AFRV) and director of the Computer Science Laboratory of INSA de Rennes.