Decoding finger movements from ECoG signals using switching linear models

*Rémi Flamary * and Alain Rakotomamonjy*

LITIS EA 4108 – INSA, Université de Rouen, Saint Etienne du Rouvray, France

One of the most interesting challenges in ECoG-based Brain-Machine Interface is movement prediction. Being able to perform such a prediction paves the way to high-degree precision command for a machine such as a robotic arm or robotic hands. As a witness of the BCI community increasing interest toward such a problem, the fourth BCI Competition provides a dataset which aim is to predict individual finger movements from ECoG signals. The difficulty of the problem relies on the fact that there is no simple relation between ECoG signals and finger movements. We propose in this paper, to estimate and decode these finger flexions using switching models controlled by an hidden state. Switching models can integrate prior knowledge about the decoding problem and helps in predicting fine and precise movements. Our model is thus based on a first block which estimates which finger is moving and another block which, knowing which finger is moving, predicts the movements of all other fingers. Numerical results that have been submitted to the Competition show that the model yields high decoding performances when the hidden state is well estimated. This approach achieved the second place in the BCI competition with a correlation measure between real and predicted movements of 0.42.

Keywords: ECoG, switching model, linear model, channel selection

1. INTRODUCTION

Some people who suffer neurological diseases can become severely impaired and have strongly reduced motor functions but still have some cognitive abilities. One of their possible way to communicate with their environment is by using their brain activities. Brain-Computer interfaces (BCI) research aim at developing systems to help such disabled people communicating with other people through machines. Non-invasive BCIs have recently received a lot of interest because of their easy protocol for sensor implantation on the scalp surface (Blankertz et al., 2004; Wolpaw and McFarland, 2004). Furthermore, although the electroencephalogram signals have been recorded through the skull and are known to have poor Signal-to-Noise Ratio (SNR), those BCI have shown great capabilities, and have already been considered for daily use by Amyotrophic Lateral Sclerosis (ALS) patients (Sellers and Donchin, 2006; Nijboer et al., 2008; Sellers et al., 2010).

However, non-invasive recordings still show some drawbacks including poor signal-to-noise ratio and poor spatial resolution. Hence, in order to overcome these issues, invasive BCI may instead be considered. For instance, Electrocorticographic recordings (ECoG) have recently received a great amount of attention owing to their semi-invasive nature as they are recorded from the cortical surface. They offer high spatial resolution and they are far less sensitive to artifact noise. Feasibility of invasive-based BCI have been proven by several recent works (Leuthardt et al., 2004; Hill et al., 2006, 2007; Shenoy et al., 2008). Yet, most of these papers consider motor imagery as a BCI paradigm and thus do not take advantage of the fine degree of control that can be gained with the ECoG signals.

Achieving such high degree of control is an important challenge for BCI since it would make possible the control of a cursor, a mouse pointer, or a robotic arm (Wolpaw et al., 1991; Krusienski et al., 2007). Toward this aim, a recent breakthrough has been made by Schalk et al. (2007) proving that ECoG recordings can lead to continuous BCI control with multiple degree of freedom. Along with the work of Pistohl et al. (2008), they have studied the problem of predicting real arm movements from ECoG signals. As such, it is important to note that unlike other BCIs, real movements are considered in these works, hence the global approaches they propose do not suit to impaired subjects.

Following the road paved by Schalk et al. (2007) and Pistohl et al. (2008), we investigate in this work, the feasibility of a fine degree of resolution in BCI control by addressing the problem of estimating finger flexions through ECoG signals. Indeed, we propose in this paper a method for decoding finger movements from ECoG data based on switching models. The underlying idea of these switching models is based on the hypothesis that movements of each of the five fingers are triggered by an internal discrete state that can be estimated and that all finger movements depend on that internal state. While such an idea of switching models have already been successfully used for arm movement prediction on monkeys, based on micro-electrode array measures (Darmanjian et al., 2006), here we develop a specific approach adapted to finger movements. The global method has been tested and evaluated on the 4th Dataset of the BCI Competition IV (Miller and Schalk, unpublished) yielding to a second place in the competition.

The paper is organized as follows: first, we briefly present the BCI Competition IV dataset used in this paper for evaluating our method and provide an overview of the global decoding method. Then we delve into the details of the proposed switching models for finger movement prediction from ECoG signals. Finally, we present numerical experiments designed for evaluating our contribution followed by a discussion about the limits of our approach and about future works.

2. DATASET

In this work, we focus on the fourth dataset from the BCI Competition IV (Miller and Schalk, unpublished). The task related to this dataset is to predict finger flexions and finger extensions from signals recorded on the surface of the brain of the subjects. In what follows, we use the term "flexion" for both kinds of movements as in the dataset description (Miller and Schalk, unpublished). The signals composing the dataset, have been acquired from three epileptic patients who had platinium electrode grids placed on the surface of their brain, the number of electrodes varying between 48 and 64 depending on the subject. Note that the electrode positions on the cortex have not been provided by the competition organizer impeding thus the use of spatial prior knowledge such as electrode's neighborhood for building the finger movement model.

Electrocorticographic (ECoG) signals of the subjects were recorded at a 1-kHz sampling using BCI2000 (Schalk et al., 2004). A band-pass filter from 0.15 to 200 Hz was applied to the ECoG signals. The finger flexion of the subject was recorded at 25 Hz and up-sampled to 1 kHz. Note that the finger flexion signals have been normalized with zero mean an unit variance. Due to the acquisition process, a delay appears between the finger movement and the measured ECoG signal. To correct this time-lag, we apply a 37-ms delay to the ECoG signals (Miller and Schalk, unpublished) as suggested in the dataset description.

The full BCI Competition dataset consists in a 10-min recording per subject. Six minutes 40 s (400,000 samples) were given to the contestants for learning the finger movement models and the remaining 3 min 20 s (200,000 samples) were used for evaluating the model. Since the finger flexion signals have been up-sampled and thus are partly composed of artificial samples, we have down-sampled the number of points by a factor of 4 leading to a training set of size 100,000 and a testing set of size 50,000. The 100,000 samples provided for learning have been split in a training (75,000) and validation set (25,000). Note that all parameters in the proposed approach have been selected in order to minimize the error on the validation set. All results presented in the paper have been obtained on the testing set provided by the competition.

In this competition, performances of methods proposed by competitors were evaluated according to the correlation between measured and predicted finger flexions. These correlations were averaged across fingers and across subjects to obtain the overall method performance. However, because its movements were highly physically correlated with those of the other finger, the fourth finger is not included in the evaluation (Miller and Schalk, unpublished).

3. FINGER FLEXION DECODING USING SWITCHING LINEAR MODELS

This section presents our decoding method for addressing the problem of estimating finger movement from ECoG signals. At first, we introduce a global view of the model and briefly discuss the decoding scheme. The second part of the section is devoted to a detailed description of each block of the switching model: the internal state estimation, the linear finger prediction, and the final decoding stage.

3.1. OVERVIEW

The main idea on which our switching model used for predicting finger movement from ECoG signals is built, is the assumption that measured ECoG signals and finger movements are intrinsically related to one or several internal states.

These internal states play a central role in our model. Indeed, it allows us to learn from training examples a specific model associated to each single state. In this sense, the complexity of our model depends on the number of possible hidden states: if only one hidden state is possible then all finger movements would be predicted by a single model. If more hidden states are allowed, then we can build more specific models related to specific states of the ECoG signals. According to this, allowing too many hidden states may thus lead to a global model that overfits the training data.
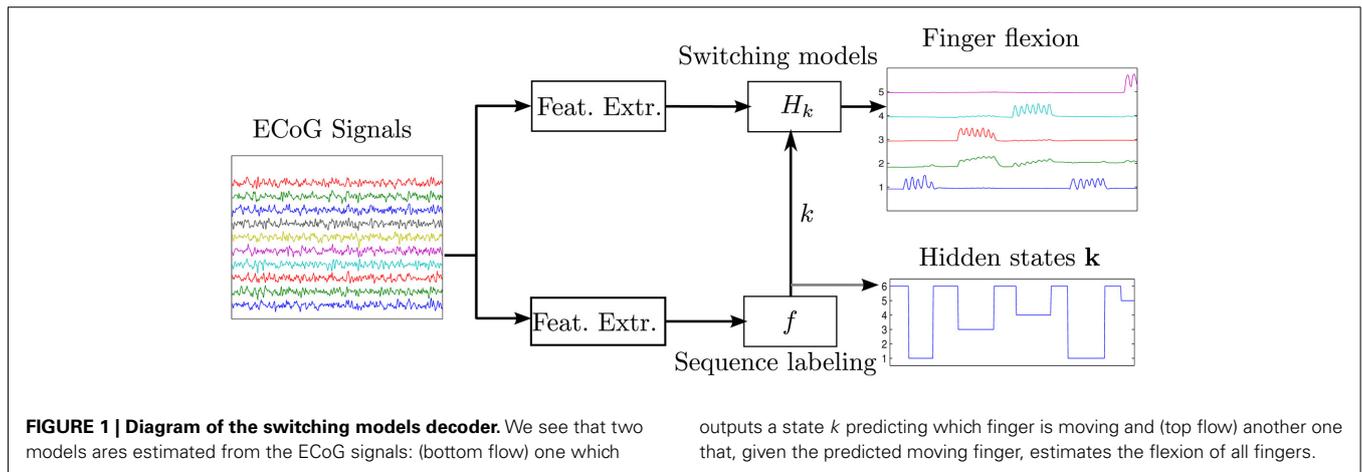
In the learning problem addressed in here, the ECoG signals present some specificities. Indeed, during the acquisition process, subjects have been dictated to move only one finger at a time, thus it appeared natural to us to take profit of this prior knowledge for building a better model. Hence, we have considered an internal state k that can take six different values, depending on which finger is moving: $k = 1$ for the thumb to $k = 5$ for the baby finger or $k = 6$ for no finger movement. This is in accordance with the experimental set-up where mutually exclusive states are in play, nonetheless for another dataset where any number of finger can move simultaneously, we can use another model for the hidden states, e.g., one binary state per finger corresponding to movement or no movement.

Figure 1 summarizes the big picture of our finger movement decoding scheme. Basically, the idea is that based on some features extracted from the ECoG signals, the internal hidden state triggering the switching finger models can be estimated. Then, this state k controls a linear switching model of parameters $\mathbf{H}_k(\tilde{\mathbf{x}})$ that will predict the finger movements.

According to this global scheme, we need to estimate the function $f(\cdot)$ that maps the ECoG features to an internal state $k \in \{1, \ldots, 6\}$ and estimate the parameter of the linear model $\mathbf{H}_k(\cdot)$ that relates the brain signals to the movements of all fingers. The next paragraphs introduce these functions and clarify how they have been learned from training data.

3.2. MOVING FINGER ESTIMATION

The proposed switching model requires the estimation of an hidden state. In our application, the hidden state k is a discrete state representing which finger is moving. Learning a model $f(\cdot)$ that estimates these internal states can be interpreted as a sequence labeling problem. There exists several sequence labeling approaches in the literature such as HMM or CRF. Nevertheless,

FIGURE 1 | Diagram of the switching models decoder. We see that two models ares estimated from the ECoG signals: (bottom flow) one which outputs a state k predicting which finger is moving and (top flow) another one that, given the predicted moving finger, estimates the flexion of all fingers.

those methods require an offline decoding, typically based on the well-known Viterbi algorithm, which precludes their applications for online movement prediction. We propose in the sequel to use a simple time sample classification scheme for solving this sequence labeling problem.

In the following, we depict the features and the methodology used for learning the function $f(\cdot)$ that predicts the value of this internal state.

Feature extraction

We used smoothed Auto-Regressive (AR) coefficients of the signal as features because they capture some dynamics of the signals. The global overview of the feature extraction procedure is given in **Figure 2**. For a single channel, the procedure is the following. The signal is divided in non-overlapping windows of 300 samples. For each window, an auto-regressive model is estimated. Thus, AR coefficients are obtained at every 300 samples (denoted by the vertical dashed line and the cross in **Figure 2**). In order to have continuous values of the AR coefficients, a smoothing spline-based interpolation between two consecutive AR coefficients is applied. Note that instead of interpolating, we could have computed the AR coefficients at each time instant, however, this heuristic has the double advantage of being computationally less demanding and of providing some smoothed (and thus more robust to noise) AR coefficients. For computational reasons, only the two first AR coefficients of a model of order 10 are used as features. Indeed, we find out from a validation procedure that among all the AR coefficients, the two first ones are the most discriminative. Further signal dynamics is taken into account by applying a similar procedure to shifted versions of the signal at $(+t_s$ and $-t_s)$, which multiplies the number of features by 3. Note that by using a positive lag $+t_s$, our feature extraction approach becomes non-causal which in principle, would preclude its use for real-time BCI. Nevertheless, this lag does not exceed the window size used for AR feature extraction and thus it has limited impact on the delay of the decision process, while it considerably enhances the system performances.

To summarize, for measurements involving 48 channels, the feature vector at a time instant t is obtained by concatenating the six AR features extracted from each single channel, leading to a resulting vector of size $48 \times 3 \times 2 = 240$.

Channel selection

Actually, some channels are not used in the function $f(\cdot)$, since we decided to perform a simple channel selection to keep only the most relevant ones. For us, the channel selection has two objectives: (i) to substantially reduce the number of channels and thus to minimize the computational effort needed for estimating and evaluating the function $f(\cdot)$ and (ii) to keep in the model only the most relevant channels so as to increase estimation performances. For this channel selection procedure, the feature vector $\mathbf{x}_t$ at time t has been computed as described above, except that for computational reasons, we do not have considered the shifted signal versions and use only the first AR coefficient. Again, we experienced on the validation set that these features were sufficient for having a good estimate of the relevant channels.

Then, for each finger, we estimate a linear regression of the form $\mathbf{x}^t \mathbf{c}_k$, based on the training set $\{\mathbf{x}_t, y_t\}$, where $\mathbf{x}_t \in \mathbb{R}^{chan}$ is a feature vector of number of channels dimension and y_t takes values $+1$ or -1 whether the considered finger is moving or not. Once the coefficient vectors $\mathbf{c}_k$ for all finger are estimated, we compute the relevance score of each channel as:

$$\mathbf{s} = \sum_{k=1}^{6} |\mathbf{c}_k| \quad \text{with } \mathbf{s} \in \mathbb{R}^{chan}$$

where the absolute value is applied elementwise. The M relevant channels are those having the M largest score in the vector $\mathbf{s}$, M being chosen in order to maximize the correlation on the validation set. This approach, although unusual, is highly related to a channel selection method based on a mixed-norm. Indeed, the above criterion can be understood as a $\ell_0 - \ell_1$ criterion where the ℓ_0 selection would have been performed by comparing channels ℓ_1 scores to an adaptive threshold dependent on M. Note that this channel selection scheme has also been successfully used for sensor selection in other competitions (Labbé et al., 2010).

Model estimation

The procedure for learning the function $f(\cdot)$ is as follows. First, since in this particular problem, the finger movements are
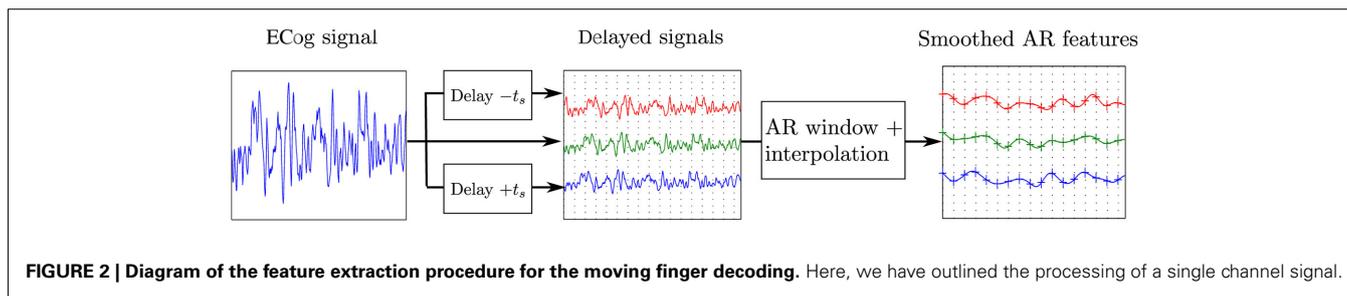
FIGURE 2 | Diagram of the feature extraction procedure for the moving finger decoding. Here, we have outlined the processing of a single channel signal.

mutually exclusive, we consider a winner-takes-all strategy and defines $f(\cdot)$ as:

$$f(\mathbf{x}) = \underset{k=1,\ldots,6}{\mathrm{argmax}}\ f_k(\mathbf{x}) \qquad (1)$$

where $f_k(\cdot)$ are linear real-valued functions of the form $f_k(\mathbf{x}) = \mathbf{x}^T \mathbf{c}_k$, that are trained so as to predict the presence or absence of finger movement for finger k. The features we use take into account some dynamics of the ECoG signals through the shifted features ($+t_s$ and $-t_s$) and a finer feature selection has been performed by means of a simultaneous sparse approximation method as described in the sequel.

Let us consider the training examples $\{\mathbf{x}_t, \mathbf{y}_t\}_{t=1}^{\ell}$ where $\mathbf{x}_t \in \mathbb{R}^d$, with $d = 240$ and $y_{t,k} = \{1, -1\}$, being the k-th entry of vector $\mathbf{y}_t \in \{1, -1\}^6$, t denoting the time instant and k denoting the internal states of each finger. $y_{t,k}$ tells us whether the finger $k = 1, \ldots, 5$ is moving at time t while $y_{t,6} = 1$ translates the fact that no fingers are moving at time t. Now, let us define the matrices $\mathbf{Y}, \mathbf{X}$, and $\mathbf{C}$ as:

$$[\mathbf{Y}]_{t,k} = y_{t,k} \quad [\mathbf{X}]_{t,j} = x_{t,j} \quad [\mathbf{C}]_{j,k} = c_{j,k}$$

where $x_{t,j}$ and $c_{j,k}$ are the j-th components of respectively $\mathbf{x}_t$ and $\mathbf{c}_k$. The aim of simultaneous sparse approximation is to learn the coefficient matrix $\mathbf{C}$ while yielding the same sparsity profile in the different finger models. The task boils down to the following optimization problem:

$$\hat{\mathbf{C}} = \underset{\mathbf{C}}{\mathrm{argmin}}\ \| \mathbf{Y} - \mathbf{XC} \|_F^2 + \lambda_s \sum_i \| \mathbf{C}_{i,\cdot} \|_2 \qquad (2)$$

where λ_s is a trade-off parameter that has to be properly tuned and $\mathbf{C}_{i,\cdot}$ is the i-th row of $\mathbf{C}$. Note that our penalty term is a mixed $\ell_1 - \ell_2$ norm similar to the one used for group-lasso (Yuan and Lin, 2006). Owing to the ℓ_1 penalty on the ℓ_2 row-norm, such a penalty tends to induce row-sparse matrix $\mathbf{C}$. Problem (2) has been solved using the block-coordinate descent algorithm proposed by Rakotomamonjy (2011).

3.3. LEARNING FINGER FLEXION MODELS

We now discuss the model relating the ECoG signal and the finger movement amplitudes. This model is controlled by an internal state k, which means that we build an estimation of the movement of all fingers and the choice of the appropriate model then depends on the estimated internal state $\hat{k}$. Hence, for each value of the internal state k, we learn a linear model $g_{k,j}(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^T \mathbf{h}_j^{(k)}$,

with $\mathbf{h}_j^{(k)} \in \mathbb{R}^{d'}$ being a vector of coefficients, that predicts the movement of finger $j = 1, \ldots, 5$ when the finger k is moving. Note that obviously the movements of the fingers $j \neq k$ are quite small but different from zero as the finger movements are physically correlated. Linear model has been chosen since they proved to achieve good performances for decoding movements from ECoG signals (Schalk et al., 2007; Pistohl et al., 2008).

Feature extraction is performed following the same line as (Pistohl et al., 2008), i.e., we use filtered time samples as features. First, all channels are filtered with a Savitsky-Golay (third order, 0.4 s width) low-pass filter. Then, the feature vector at time t, $\tilde{\mathbf{x}}_t \in \mathbb{R}^{d'}$ is obtained by concatenating the time samples at t, $t - \tau$, and $t + \tau$ for all smoothed signals and for all channels. Samples at $t - \tau$ and $t + \tau$ are used in order to take into account slight temporal delays between the brain activity and finger movements. Once more, our decoding method is thus not causal, but as only small values for τ are considered ($\ll 1$ s), the decision delay is reduced.

Let us now detail how the matrix $\mathbf{H}_k = [\mathbf{h}_1^{(k)} \cdots \mathbf{h}_5^{(k)}] \in \mathbb{R}^{d' \times 5}$ containing the finger linear model parameters for state k, is learned. For each finger k, we extract all samples $\tilde{\mathbf{x}}_t$ when that finger is known to be moving. For this purpose, we have manually segmented the signals and extracted the appropriate signal segments, in order to build the target matrix $\tilde{\mathbf{Y}}_k$ (finger movement amplitude when finger k is moving), and the corresponding feature matrix $\tilde{\mathbf{X}}_k$. This training sample extraction stage is illustrated on **Figure 3**.

Finally, the linear models for state $k \in 1, \ldots, 5$ are learned by solving the following multi-dimensional ridge regression problem:

$$\min_{\mathbf{H}_k} \| \tilde{\mathbf{Y}}_k - \tilde{\mathbf{X}}_k \mathbf{H}_k \|_F^2 + \lambda_k \| \mathbf{H}_k \|_F^2 \qquad (3)$$

with λ_k a regularization parameter that has to be tuned.

For this finger movement estimation problem, we also observed that feature selection helps in improving performances. Again, we use the estimated matrix $\hat{\mathbf{H}}$ coefficients for pruning the model, $\hat{\mathbf{H}}$ being the minimizer of equation (3). Similarly to the feature selection procedure introduced in section 2, we keep the M' features having the largest entries of vector $\sum_{i=1}^{5} | \hat{\mathbf{h}}_i^{(k)} |$. M' being chosen as to minimize the validation error.

3.4. DECODING FINGER MOVEMENT

Once the linear models and the hidden state estimator are learned, we can apply the decoding scheme given in **Figure 1**. The decoding is a 2-step approach requiring the two feature vectors $\mathbf{x}_t$ and $\tilde{\mathbf{x}}_t$ used respectively for moving finger estimation and for estimating
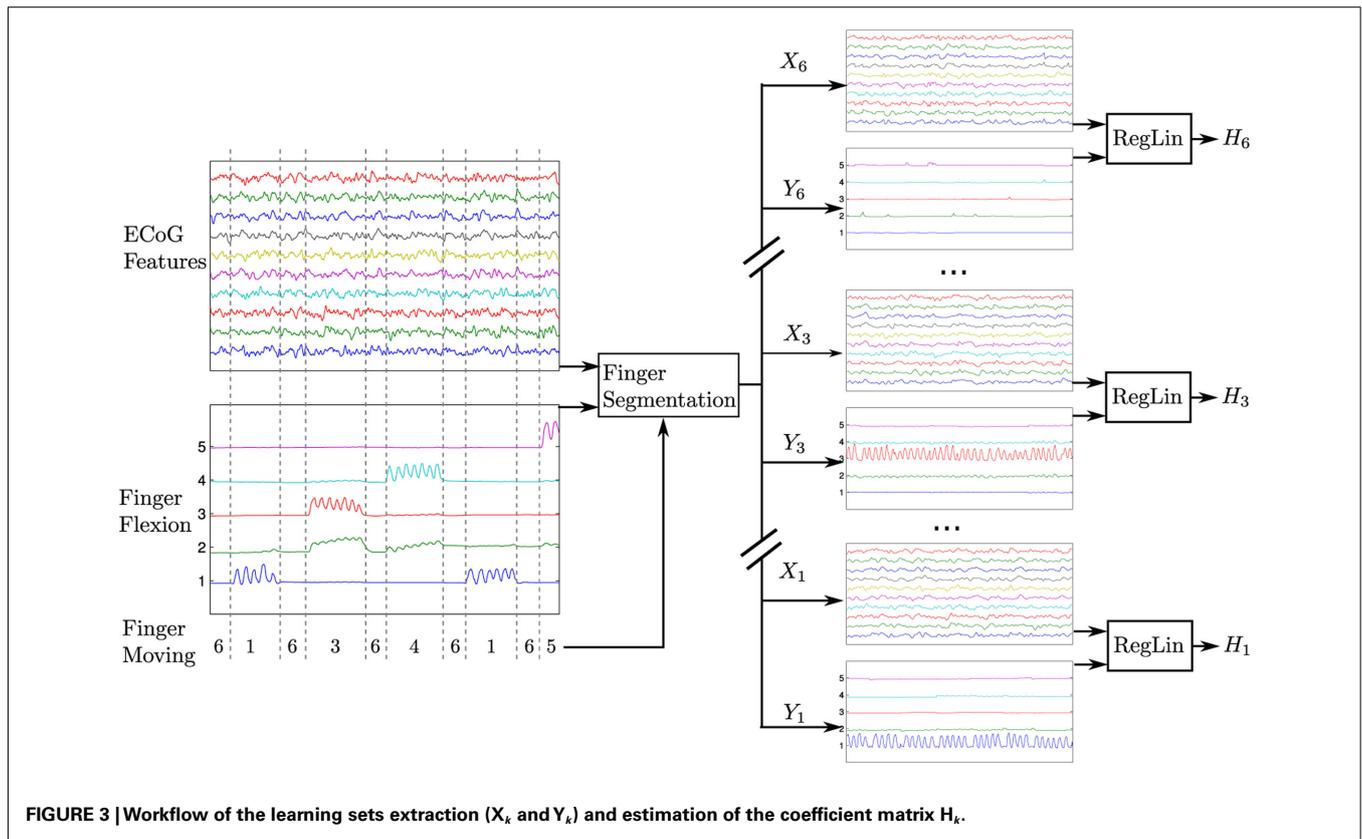
FIGURE 3 | Workflow of the learning sets extraction (X_k and Y_k) and estimation of the coefficient matrix H_k.

all finger flexion amplitudes. Estimated finger positions at time t is obtained by:

$$\hat{\mathbf{y}}_t = \tilde{\mathbf{x}}_t^T \hat{\mathbf{H}}_{\hat{k}} \qquad \text{with } \hat{k} = \underset{k}{\arg\max} \quad \mathbf{x}_t^t \mathbf{c}_k \qquad (4)$$

with $\hat{\mathbf{y}}_t \in \mathbb{R}^5$ a vector containing the estimated finger movements at time t, $\hat{k}$ the estimated moving finger, and $\hat{\mathbf{H}}_{\hat{k}}$ the estimated linear model for state k.

4. RESULTS

In this section, we discuss the performances of our switching model decoder. But at first, we explain how the different parameters the overall model have been selected. Next, we establish the soundness of linear models for finger movement prediction by evaluating their performance just on part of the test examples related to movements. Finally, we evaluate our approach on the complete data and compare its performances to those of other competitors.

4.1. PARAMETER SELECTION

All parameters used in the global model are selected by a validation method on the last part of the training set (75,000 for the training, 25,000 for validation). We suppose that the validation set size is large enough to avoid over-fitting. Examples of training and validation set sizes for each hidden state k are given in **Table 1** for subject 1.

Hence, for the learning function $f(\cdot)$, we select the number of relevant channels M, the time-lag t_s used in feature extraction

Table 1 | Number of samples used in the validation step for subject 1.

Finger	Learning	Validation
1	8355	3848
2	9750	2965
3	13794	3287
4	6179	2915
5	10729	5362
6	26074	6623

and the regularization term λ_s of equation (2) while for estimating $\mathbf{H}_k$, we tune the number of selected channels M', τ, and λ_k. All these parameters are chosen so that they optimize the model performances on the validation set.

4.2. LINEAR MODELS H_K FOR PREDICTING MOVEMENT

Linear models are known to be accurate models for arm movement prediction (Schalk et al., 2007; Pistohl et al., 2008), here we want to confirm the hypothesis that their use can also be extended to finger movement predictions. For each finger k, we extract the ECoG signals and finger flexion amplitude when that finger is actually moving (see **Figure 4**) and we predict the finger movement on the test examples using the appropriate column of $\mathbf{H}_k$. The correlation between the predicted finger movement $\tilde{\mathbf{X}}_k \hat{\mathbf{h}}_k^k$ and the actual movement $\mathbf{y}_k$ is computed for each finger k of each subject. The results are shown in **Table 2** and since they have been obtained only on samples where the actual finger is moving, they can not
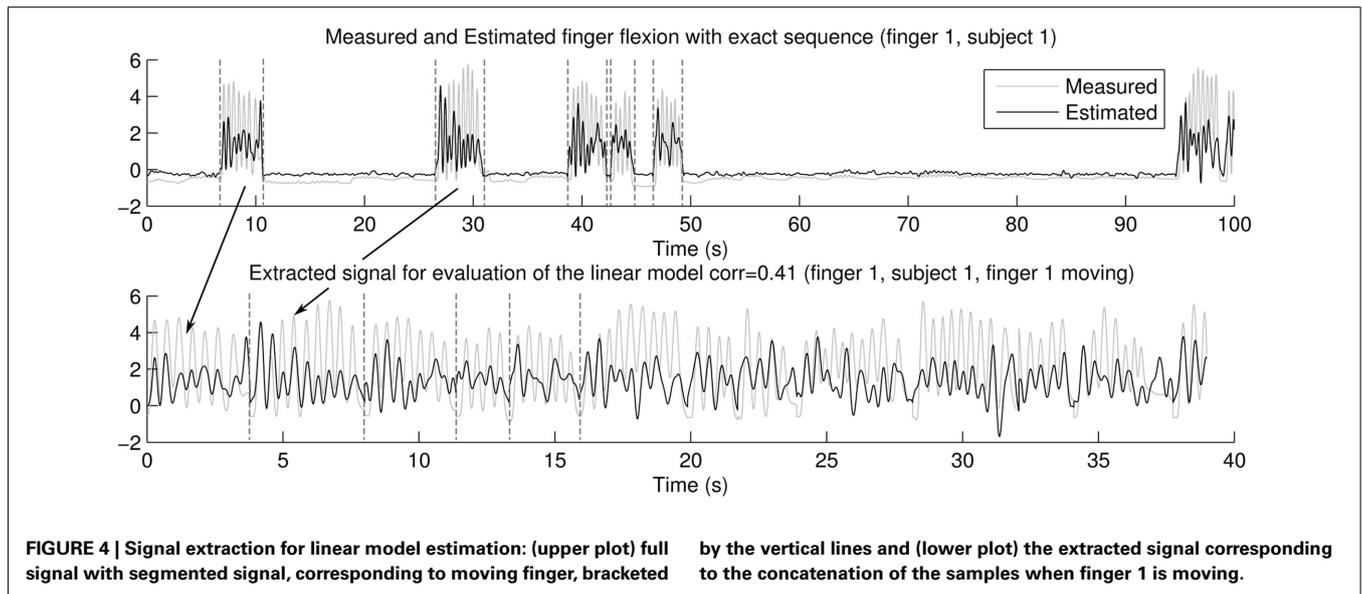
FIGURE 4 | Signal extraction for linear model estimation: (upper plot) full signal with segmented signal, corresponding to moving finger, bracketed by the vertical lines and (lower plot) the extracted signal corresponding to the concatenation of the samples when finger 1 is moving.

Table 2 | Correlation coefficient obtained by the linear models $\mathbf{h}_k^{(k)}$.

Finger	Sub. 1	Sub. 2	Sub. 3	Average
1	0.4191	0.5554	0.7128	0.5625
2	0.4321	0.4644	0.6541	0.5169
3	0.6162	0.3723	0.2492	0.4126
4	0.4091	0.5668	0.0781	0.3513
5	0.4215	0.5165	0.5116	0.4832
Avg.	0.4596	0.4951	0.4411	0.4653

be compared to the competition results as they do not take into account part of the signals related to other finger movements.

We observe that by using a linear regression between the feature extracted from the ECoG signals and the finger flexions, we achieve a correlation of 0.46 (averaged across fingers and subjects). This results correspond to those obtained for the arm trajectory prediction (Schalk et al., 2007 obtained 0.5 and Pistohl et al., 2008 obtained 0.43). We can then conclude that the linear models provide an interesting baseline for finger movement predictions.

4.3. SWITCHING MODELS FOR MOVEMENT PREDICTION

In order to evaluate the accuracy and the contributions of our the switching model, we report three different results: (a) we compute the estimated finger flexion using a unique linear model trained on all samples (including those ones where the considered finger is not moving), (b) we decode finger flexions with our switching decoder while assuming that the exact sequence of hidden states is known[1], and (c) we use the proposed switching decoder with the estimated hidden states.

Correlation coefficients between real flexions and those predicted by the baseline model (a) are reported in **Table 3A** and predicted flexions can also be seen on the upper plots of **Figure 5**. We note that the correlations obtained are rather low (an average

of 0.30). We conjecture that this is due to the fact that the model parameters are learned on the complete signal (which includes no movement). Indeed, the long temporal segments with small magnitudes have a tendency to shrink the global output toward zero. This is an issue that can be addressed by using our switching linear models.
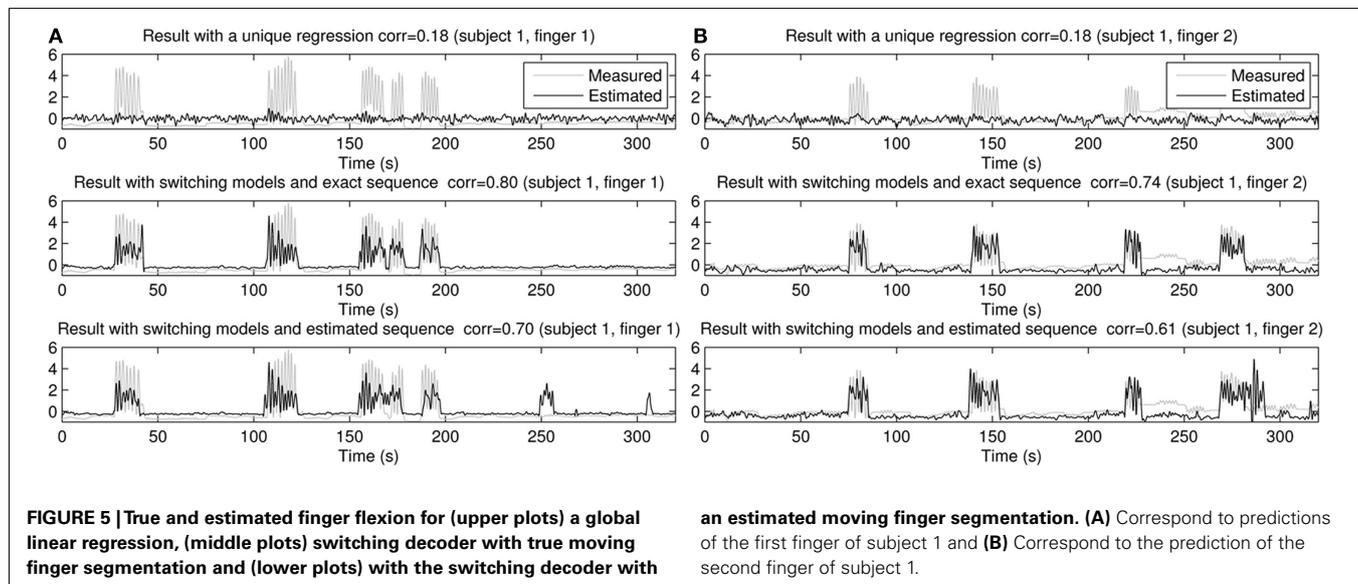
The switching model decoder is a two-part process as it requires the linear models $\mathbf{H}_k$ and the sequence of hidden states (see **Figure 1**). In order to evaluate the optimal performances of the switching model, we apply the decoder using the exact sequence $\mathbf{k}$ obtained from the actual finger flexion. We know that this cannot be done in practice as it would imply a perfect sequence labeling, but in our opinion, it gives an interesting idea of the potential of the switching models approach for given linear models $\mathbf{H}_k$. Examples of estimation can be seen in the middle plots of **Figure 5** and while correlation coefficients are in **Table 3B**. We obtain a high accuracy across all subjects with an average correlation of 0.61 when using an exact sequence. This proves that the switching model can be efficiently used for decoding ECoG signals.

Finally, we evaluate the switching models approach when using the finger moving estimator. In other words, we use the switching models $\mathbf{H}_k$ to decode the signals with equation (4) and the estimated sequence $\hat{\mathbf{k}}$. The finger movement estimation can be seen on the lower plot of **Figure 5B** and the correlation measures are in **Table 3C**. As expected, the accuracy is lower than those

---

[1] This is possible since the finger movements on the test set are now available.

Table 3 | Correlation between measured and estimated movement for a global linear regression (A), switching decoder with exact sequence (B), and switching decoder with an estimated sequence (C).

Finger	Sub. 1	Sub. 2	Sub. 3	Average
(A) LINEAR REGRESSION				
1	0.1821	0.2604	0.3994	0.2807
2	0.1844	0.2562	0.4247	0.2884
3	0.1828	0.2190	0.4607	0.2875
4	0.2710	0.4225	0.5479	0.4138
5	0.1505	0.2364	0.3765	0.2545
Avg.	0.1942	0.2789	0.4419	0.3050
(B) SWITCHING MODELS (EXACT SEQUENCE)				
1	0.8049	0.5021	0.8030	0.7033
2	0.7387	0.4638	0.7655	0.6560
3	0.7281	0.4811	0.7039	0.6377
4	0.7312	0.5366	0.6241	0.6307
5	0.2296	0.4631	0.6126	0.4351
Avg.	0.6465	0.4893	0.7018	0.6126
(C) SWITCHING MODELS (EST. SEQUENCE)				
1	0.7016	0.3533	0.6457	0.5669
2	0.6129	0.3045	0.5097	0.4757
3	0.2774	0.0043	0.4025	0.2280
4	0.4576	0.2782	0.5920	0.4426
5	0.3597	0.2507	0.6553	0.4219
Avg.	0.4818	0.2382	0.5611	0.4270

FIGURE 5 | True and estimated finger flexion for (upper plots) a global linear regression, (middle plots) switching decoder with true moving finger segmentation and (lower plots) with the switching decoder with an estimated moving finger segmentation. **(A)** Correspond to predictions of the first finger of subject 1 and **(B)** Correspond to the prediction of the second finger of subject 1.

obtained with the true segmentation. However, we obtained an average correlation of 0.42 which is far better than the correlation obtained from a unique linear model. These predictions of the finger flexions were presented in the BCI Competition and achieved the second place. Note that the last 3 fingers have the lowest performances. Indeed, those fingers are highly physically correlated and much more difficult to discriminate than the two first ones in the sequence labeling. The first finger is by far the best estimated one as we obtained a correlation averaged across subject of 0.56.

DISCUSSION AND FUTURE WORKS

The results presented in the previous section have been submitted to the BCI Competition IV. We achieved the second place with an average correlation of 0.42 while the best performance have been obtained obtained by Liang and Bougrain, 2009; a correlation of about 0.46). Their method considers an amplitude modulation along time to cope with the abrupt change in the finger flexions magnitude along time. Such an approach is somewhat similar to ours since they try to distinguish between situations where fingers were moving or fingers were still.

We believe that our approach can be improved in several ways. Indeed, we chose to use linear models that are triggered by an internal state, while (Pistohl et al., 2008) proposed to use a Kalman filter for the movement decoding. Hence, it would be interesting to investigate whether Kalman filter or a non-linear model can help us in getting a better model.

Furthermore, our sequence labeling approach for estimating the sequence of hidden states can be also improved. Liang and Bougrain (2009) proposed to use Power Spectral Densities of the ECoG channel as features and we believe that the sequence labeling might benefit from the use of this kind of features. Finally, we have used a simple sequence labeling approach by doing a temporal sample classification of low-pass filtered features. Since other sequence labeling methods like Hidden Markov Models (Darmanjian et al., 2006) or Conditional Random Fields (Luo and Min, 2007) have been successfully proposed for BCI application, we believe that a more robust sequence labeling approach can increase the quality of the estimated segmentation and thus the final performances. For instance, the sequence SVM proposed by Bordes et al. (2008) can efficiently decode a sequence in real-time and has shown good predictive performances.

Finally, the question of how to predict statically held finger position is still open. Our approach has shown good finger movement estimations but when still, fingers were always at the same resting position, which is a favorable case. To address this problem of held finger, we can simply extend our global model by making the internal state estimator predict a static finger position.

CONCLUSION

In this paper, we present a method for finger flexions prediction from ECoG signals. The decoder, based on switching linear models, has been evaluated in the BCI Competition IV Dataset 4 and achieved the second place in the competition. We show empirically the advantages of the switching models scheme over a unique model. Finally the results suggest that the model performances are highly dependent on the hidden state estimation accuracy. Hence, improving this estimation would naturally imply an overall performance improvement.

In future works, we plan to improve the results of the switching models decoder by two different approaches. On the one hand, we want to investigate the usefulness of more general models than linear ones for the movement prediction (switching kalman filters, non-linear regression). On the other hand, we can improve the moving finger decoding step using other sequence labeling approaches or by considering other features extracted from the ECoG signals.

ACKNOWLEDGMENTS

REFERENCES

Blankertz, B., Muller, K.-R., Curio, G., Vaughan, T. M., Schalk, G., Wolpaw, J. R., Schlogl, A., Neuper, C., Pfurtscheller, G., Hinterberger, T., Schroder, M., and Birbaumer, N. (2004). The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials. *IEEE Trans. Biomed. Eng.* 51, 1044–1051.

Bordes, A., Usunier, N., and Bottou, L. (2008). "Sequence labelling SVMs trained in one pass," in *Machine Learning and Knowledge Discovery in Databases: ECML PKDD 2008*, eds W. Daelemans, B., Goethals, and K. Morik (Springer: Lecture Notes in Computer Science, LNCS 5211), 146–161.

Darmanjian, S., Kim, S.-P., Nechyba, M. C., Principe, J., Wessberg, J., and Nicolelis, M. A. L. (2006). "Independently coupled HMM switching classifier for a bimodel brain-machine interface," in *Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, 379–384.

Hill, N., Lal, T., Schroeder, M., Hinterberger, T., Wilhem, B., Nijboer, F., Mochty, U., Widman, G., Elger, C., Scholkoepf, B., Kuebler, A., and Birbaumer, N. (2006). Classifying EEG and ECoG signals without subject training for fast BCI implementation: comparison of non-paralysed and completely paralysed subjects. *IEEE Trans. Neural Syst. Rehabil. Eng.* 14, 183–186.

Hill, N., Lal, T., Tangermann, M., Hinterberger, T., Widman, G., Elger, C., Scholkoepf, B., and Birbaumer, N. (2007). "Toward brain-computer interfacing," in *Classifying Event-Related Desynchronization in EEG, ECoG and MEG signals*, eds G. Dornhege, J. R. Milan, T. Hinterberger, D. McFarl, and K. R. Müller (MIT Press), 235–260.

Krusienski, D. J., Schalk, G., McFarland, D. J., and Wolpaw, J. R. (2007). A mu-rhythm matched filter for continuous control of a brain-computer interface. *IEEE Trans. Biomed. Eng.* 54, 273–280.

Labbé, B., Tian, X., and Rakotomamonjy, A. (2010). "MLSP competition, 2010: description of third place method," in *2010 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Kitilla, 116–117.

Leuthardt, E., Schalk, G., Wolpaw, J., Ojemann, J., and Moran, D. (2004). A brain-computer interface using electrocorticographic signals in humans. *J. Neural Eng.* 1, 63–71.

Liang, N., and Bougrain, L. (2009). "Decoding finger flexion using amplitude modulation from band-specific ECoG," in *European Symposium on Artificial Neural Networks – ESANN*, Bruges.

Luo, G., and Min, W. (2007). Subject-adaptive real-time sleep stage classification based on conditional random field. *AMIA Annu Symp Proc.*, Chicago.

Nijboer, F., Sellers, E., Mellinger, J., Jordan, M., Matuz, T., Furdea, A., Mochty, U., Krusienski, D., Vaughan, T., Wolpaw, J., Birbaumer, N., and Kubler, A. (2008). A brain-computer interface for people with amyotrophic lateral sclerosis. *Clin. Neurophysiol.* 119, 1909–1916.

Pistohl, T., Ball, T., Schulze-Bonhage, A., Aertsen, A., and Mehring, C. (2008). Prediction of arm movement trajectories from ECoG-recordings in humans. *J. Neurosci. Methods* 167, 105–114.

Rakotomamonjy, A. (2011). Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms. *Signal Process* 22, 1307–1320.

Schalk, G., McFarland, D. J., Hinterberger, T., Birbaumer, N., and Wolpaw, J. R. (2004). BCI2000: a general-purpose brain-computer interface (BCI) system. *IEEE Trans. Biomed. Eng.* 51, 1034–1043.

Schalk, G, Kubanek, J., Miller, K. J., Anderson, N. R., Leuthardt, E. C., Ojemann, J. G., Limbrick, D., Moran, D., Gerhardt, L. A., and Wolpaw, J. R. (2007). Decoding two-dimensional movement trajectories using electrocorticographic signals in humans. *J. Neural Eng.* 4, 264–275.

Sellers, E., and Donchin, E. (2006). A P300-based brain-computer interface: initial tests by ALS patients. *Clin. Neurophysiol.* 117, 538–548.

Sellers, E., Vaughan, T., and Wolpaw, J. (2010). A brain-computer interface for long-term independent home use. *Amyotroph. Lateral Scler.* 11, 449–455.

Shenoy, P., Miller, K., Ojemann, J., and Rao, R. (2008). Generalized features for electrocorticographic BCI. *IEEE Trans. Biomed. Eng.* 55, 273–280.

Wolpaw, J. R., and McFarland, D. J. (2004). Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proc. Natl. Acad. Sci. U.S.A.* 101, 17849–17854.

Wolpaw, J. R., McFarland, D. J., Neat, G. W., and Forneris, C. A. (1991). An EEG-based brain-computer interface for cursor control. *Electroencephalogr. Clin. Neurophysiol.* 78, 252–259.

Yuan, M., and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. B* 68, 49–67.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.