31st Annual International Conference of the IEEE EMBS
Minneapolis, Minnesota, USA, September 2-6, 2009

Overcoming Measurement Time Variability in Brain Machine Interface

B. Vikrham Gowreesunker, Ahmed H. Tewfik, Vijay A. Tadipatri, Nuri F. Ince, James Ashe, Giuseppe Pellizzer
University of Minnesota, Minneapolis, USA

Abstract— **We introduce a subspace learning approach for multi-channel Local Field Potentials (LFP), and demonstrate its application in movement direction decoding for 8 directions movement. We show that the subspace learning method can effectively address the issue of signal instability across recording sessions by extracting recurrent features from the data. We present results for movement direction decoding, where we trained on two recording sessions, and evaluated decoding performance on a third session. We combine our method with a classifier based on Error-Correcting Output Codes (ECOC) and Common Spatial Patterns (CSP) and found improvement in Decoding Power (DP) from 76% to 88% for a subject known to have strong inter-session variability. Furthermore, we saw an increase from 86% to 90% DP with another subject which exhibited significantly less variability.**

I. Introduction

Different types of signals have been used in brain studies as candidate for brain machine interface (BMI) design. The small extracellular potentials generated by neurons in the cortical layers can be inferred from electroencephalogram (EEG) signals acquired non-invasively on the scalp surface, electrocorticogram (ECoG) acquired with electrodes placed on the surface of the brain, single neuron activity (SUA) and local field potential (LFP) recorded with implantable electrodes. In general, the more invasive techniques, such as SUA or LFP signals, provide higher spatial resolution and frequency bandwidth than the less invasive methods. Spatial resolution can be an important factor in brain studies and BMI systems. For example, a subject using a BMI system based on EEG with relatively poor spatial resolution will need to synchronously modulate a large population of neurons to successfully use the system. Many techniques have been proposed in the literature to study brain field potential signals but all suffer from the same problem that their performance do not generalize well across recording sessions. Our primary goal here is to introduce a novel technique that address the instability and time variability challenges associated with LFP signals recorded on different days. The potential is illustrated in the context of movement direction decoding, which is of great interest for Neural Prosthetics applications.

Signal instability and time variability are due to the acute and chronic responses of the brain tissue after implantation, brain plasticity with BMI systems as the subject learns how to optimize system performance, physiological changes in a subject due to prior activity or rest periods, or context and environmental conditions. Indeed, one of the most critical challenges in processing SUA and LFP signals in BMI applications is the change in the characteristics if such signals are collected in different sessions separated by a week or more. It is our experience that currently prevalent methods such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA), fail to capture the above mentioned time variability and instabilities. In this work, we approach the problem by seeking the subspaces where the recurrent and stable features of the signals live. Once the recurrent subspaces can be identified, the signals can be factored into a projection onto the recurrent subspace and an error term which would represent the variability. Although such a decomposition can be used to several ends, in this paper we focus on how it can improve classification in movement direction decoding using LFP.

We recently introduced an Orthogonal Subspace Pursuit (OSP) algorithm for learning sparse adaptive representations and demonstrated its potential in the context of Blind Source Separation (BSS) [1]. The foundation of the OSP algorithm is a subspace clustering method that assumes signals do not group around clusters of homogeneous dimension. For example, in a 3 dimension space, data can cluster around various lines (1-D) or planes (2-D), and our proposed method can extract the subspaces of different dimensions, while a k-means approach would force feed the data into homogeneous vectors clusters. In this work, we propose using the OSP to learn recurrent features of LFP data. We use a mixing model for the multi-channel data, similar to that of BSS, to motivate factoring the data into a recurrent subspace for the signals and a variability term. Although, the results presented are for a specific classifier, the subspace decomposition method described can be combined with any other classifier. We demonstrate decoding power improvement from 76% to 88% in the context of movement direction decoding across recording sessions for a particularly challenging subject.

The remainder of the paper is organized as follows. In section II, we introduce some preliminary concepts regarding formating LFP data and movement direction decoding. In section III, we describe the sparse coding approach to learning subspaces. In section IV, we describe in detail how the subspaces can be learned from multi-channel LFP data. In section V, we illustrate the impact of subspace decomposition on decoding movement directions using LFP data. In section VI, we summarize our results.

II. Preliminaries

The LFP data discussed in this paper is arranged in 3 sessions, where session 2 was recorded one week after session

1, and session 3 was recorded one day after session 2. In each session, several multi-channels LFP trials were recorded, such that each trial correspond to a specific directional movement. Data for eight directions, evenly spaced between 0^0 and 315^0, were recorded in each session. The goal of the movement direction decoding is to estimate which movement direction is associated with a particular trial, and is typically approached by a supervised learning method, such as the classifier presented in [2]. The accuracy of the decoding is defined in terms of Decoding Power (DP), which is the ratio of correctly classified directions to the total number of directions. It should to be noted that random classification yields a DP of 12.5%. In the remainder of this section, we describe the mixing system model that motivates the pursuit of recurrent subspaces and the premise for subspace decomposition.

Let N be the number of channels in an LFP recording trial, and T be the time duration of the recording, then the i^{th} trial, is a matrix, X_i, of dimension $N \times T$. Each session, X, is a collection of K trials such that X = $\{X_i\}_{i=1}^{K}$. We assume that each trial X_i can be modeled as a mixing system, such that $X_i = H_i \odot S$, where S are the localized sources of the signals of interest (underlying neurons), and H_i is the mixing matrix which could be instantaneous, in which case the $\odot$ operator is a simple product, or H_i could be mixing filters, and the $\odot$ operator is a convolution. Such a model, commonly used in blind source separation, [3], [4], can help us reformulate our approach to classification of multi-channel LFP data. In this model, the spatial diversity of the observation is capture in the mixing matrix/filter, H_i, while the underlying source characteristics is represented by matrix, S.

We assume that for a specific type of physiological stimulus, some sources exhibit a particular behavior which should be recurrent across trials, and even recording sessions, while the observed signal instabilities are due of the state of the subject, will be less structured. By modeling the system in a mixing fashion, we can effectively break down the classification problem as 1) finding a classifier that captures the diversity of mixing matrix/filter 2) factor the sources as a recurrent subspace, capturing the features of interest, and a non-recurring one which represents the instabilities. The classifier used in this paper applies a redundant hierarchical classification strategy using Error Correction Output Codes (ECOC) [5] and linear discriminant classifiers to find a set of Common Spatial Patterns (CSP) [6] to discriminate between directions. The full details of the classifier and its merit can be found in [2]. We assume the signal of interest, S, can be factored out as a recurrent, S_R and non-recurrent, S_{NR}, portions, such that,

$$X_i = H_i \odot (S_R + S_{NR}). \quad (1)$$

By projecting the signal onto the subspace where S_R lives, we can improve the performance of our classifier.

III. Learning Subspaces using OSP

The Orthogonal Subspace Pursuit (OSP) algorithm [1] is a method for extracting inherent subspaces from data, and was originally proposed to learn sparse signal representation. The method consist of a sparse coding stage, followed by a subspace clustering step and an optional optimization step. The OSP software is available from the authors. Below, we highlight the essence of each step.

A. Sparse Coding: Orthogonal Least Square

Let vector y, be a vector of dimension $N \times 1$, and let D be a dictionary, which consists of a set of potentially overcomplete basis vectors. A sparse decomposition algorithm seeks to approximate y in terms of a minimum number of columns of D, where sparsity is referred to the number of vector from D needed to approximate y. The OLS algorithm [7] involves two steps, the first involves finding the most correlated atom (vector) from the dictionary, and the second involves a dictionary decorrelation step where the atoms that were not selected are decorrelated from previously chosen atoms. The OLS iteratively chooses vectors from the dictionary until it either has enough vectors to approximate y or it has reached a user defined maximum number of vectors.

B. Orthogonal Subspace Pursuit (OSP)

Given a set of K observation vectors each of dimension $T \times 1$, we normalize each vector to unit l_2 norm and the resulting data is our training set, $Y = \{y_i\}_{i=1}^{K}$. We initialize the dictionary as D^0 as the data Y. The algorithm has two stages, first we identify a subspace from the training data, and second we find all the training data that lives on this subspace and remove them from Y before looking for the next subspace. The OLS is used to find the sparse representation of the vectors. A summary of the algorithm is given in figure 1.

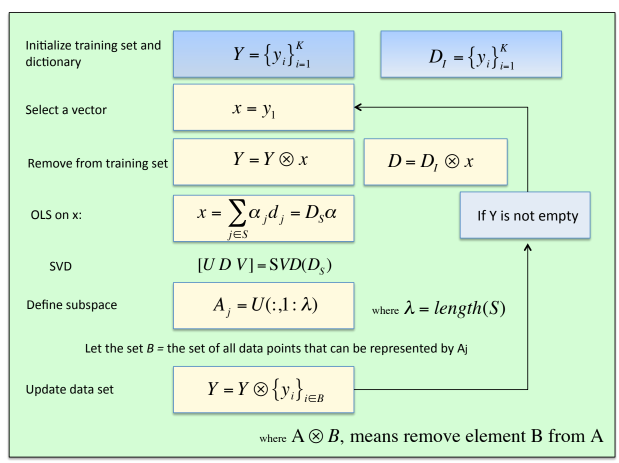

Fig. 1. Overview of the Orthogonal Subspace Pursuit algorithm

C. Subspace Optimization

Given a training set, Y, of K elements, and M subspaces, $\{A_i\}_{i=1}^{M}$, the subspace optimization step will first prune the number of subspaces to a user defined maximum, of λ_{max}, then re-optimize the subspace by iteratively re-clustering the vectors from Y over the retained subspaces. The re-clustering is done such that Y is partitioned into clusters of vectors

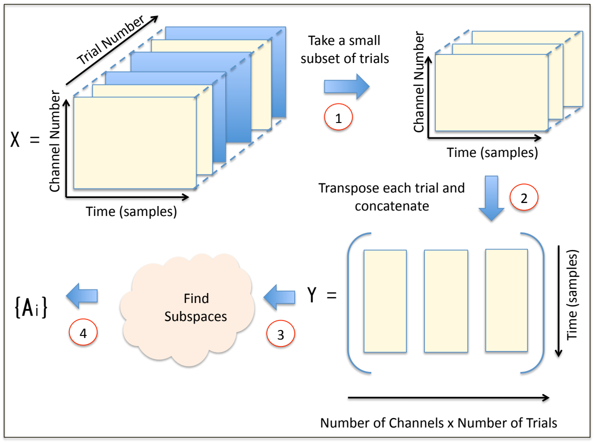

Fig. 2. Formatting Multi-channel data for subspace learning

closest to each subspace, and the subspaces are updated by doing SVD on the clustered data.

IV. Learning Recurrent Subspaces from LFP

We now describe how the subspaces are learned from multi-channel LFP data. Since our classifier is already designed to learn spatial patterns, we can focus on learning the underlying characteristics of the source signals without spatial restrictions. Hence, for our purposes, we treat each channel as a separate observation. Each recording session had about 200 trials with 128 channels, which represents an initial training set of about 50,000 vectors. To be computationally tractable, we select a random subset of trials. We first find an initial set of subspaces by applying the OSP to the subset of trials as illustrated in fig. 2, followed by a subspace pruning and optimization to remove remove subspace with very small cluster size. The procedure for each direction is as follows,

1) Select 20 random trials from the training set, $\{X_i\}_{i=1}^{20}$
2) Transpose and concatenate each trial, such that $Y = [X_1^T X_2^T \dots X_{20}^T]$
3) Apply the OSP algorithm to Y
    a) Normalize every column of Y,
    b) Find an initial set of subspaces, $\{A_i\}_{i=1}^{K}$
4) Retain a maximum of λ_{max} (typically 20) subspaces and optimize using the method of section III-C
5) Retain the 3 subspaces with largest cluster sizes
6) Repeat steps 1-4 for next direction

We found that for each direction, most of the training set clusters around 2 to 3 subspaces of less than 10 vectors each, and it is sufficient to retain only these.

A. Condensing Subspaces

We use the notion of subspace angle as described in algorithm 12.4.3 from [8] to condense the 2-3 subspaces found for each direction into a single subspace. Given two matrices, A and B, of dimension N x q and N x p respectively, using the subspace angle approach, we can find two sets of vectors, $U = \{u_i\}_{i=1}^{m}$ and $V = \{v_i\}_{i=1}^{m}$, of dimension equal to the m = minimum(p,q), that are known as the principal vectors of each subspace. Using the principal angles between vectors, we can decompose each subspace as a combination of a common subspace, and a complement. The condensed subspace is the union of the common subspace, and the complement from each subspace. This process allows us to account for signals that draws features from more than one subspace, in a computationally efficient manner.

B. Recurrence

An important parameter in finding the recurrent features is the approximation error used in the learning algorithm. Unlike applications such as speech or image processing, we do not seek to approximate the full signal. Instead, we purposely relax the approximation error to about 0.31 error norm, in the sparse coding stage, so we can capture only the most salient and recurrent features. These features are recurrent within the sessions they were trained on but can be made more robust and generalizable by considering subspaces intersection across sessions, similar to section IV-A. This extension requires more sessions for validation and is currently being studied.

V. Classification using Subspace Projection

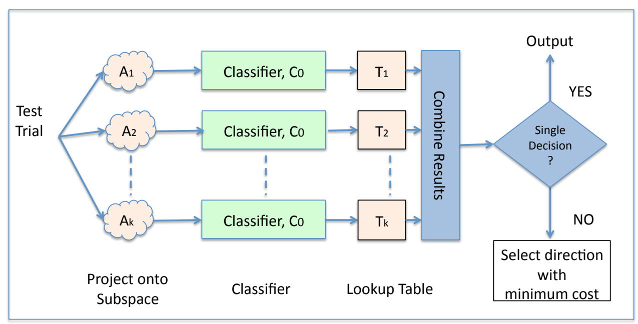

Fig. 3. Evaluating the movement direction of a trial using proposed combination of classifier and subspaces. Here the same classifier was used for all the subspaces

A. Data Acquisition and Pre-Processing

Two male rhesus monkeys were trained to perform point to point movement tasks on a horizontal manipulandum. Given a cue, the subject executed joystick movements to reach one of the eight targets. Each trial proceeded in the following way. The subject placed the joystick in the center of the manipulandum for a period of 800 ms. One of the eight peripheral targets was displayed for about 500-700 ms serving as a cue for the subject. Following a pause of about 1000 ms, the target reappeared. The subject moves the joystick and holds it at the target for about 800 ms. The order of the directions was pseudo-random and was performed in sets of eight. Our data was low passed to [0.5 Hz to 4 Hz] as this was reported to be the discriminating sub-band across all the eight directions[2]. The protocol for animal experimentation was approved by the Institutional Animal Care and Use Committee of the Veterans Affairs Medical Center (Minneapolis). The guidelines of “Public Health Service Policy on Humane Care: Use of Laboratory Animals by Awardee Institutions” and the ”NIH Principles for Use of Animals” were followed.

B. Movement Decoding Using Subspace Projections

An overview of a classification system using subspace projections is given in figure 3. Let A_k, be the k^{th} subspace of dimension $N \times M$ such that the M column vectors span the subspace A_k. C_k is the k^{th} classifier associated with projection onto the k^{th} subspace. T_k is the k^{th} row of the lookup table, T, and it filters out the directions which cannot be represented by the k^{th} subspace. Each test trial is first, separately, projected onto each subspace, and then run through a classifier, which will decide which direction it corresponds to. The decision is then matched to the lookup table which retains a decision only if that direction can be represented by the corresponding subspace. Finally, a decision rule is applied to mesh the combined decisions from each subspace projection.

C. Masking using Lookup-Table

The purpose of the lookup-table is to identify the best subspace for a particular direction. If, for example, we know that subspace, A_1 is the best subspace for the first direction (say 0^o), then we set $T(1,1) = 1$ for direction 0^o and $T(k,1) = 0$ for all other subspaces. We select only one subspace per direction, so there are only 8 non-zero entries in the matrix, T. The best subspace for a direction is the one that results in the highest DP for that direction while keeping a low false positive. The lookup table can be estimated by cross-validation from the training data.

D. Decision Rule

Ideally, for each test trial, there would be only a single non-zero decision, and that decision would be retained as the decoded direction. However, in practice, about 20 to 30% of the time, we get ambiguous decisions, where more than one subspace will give a decision. In these cases, we need to apply a more elaborate decision rule to resolve the confusion. In our case, the CSP+ECOC classifier outputs a cost associated with each decision and the lower the cost, the more confident the decision. We use a hierarchical decision process, where we first evaluate how many subspaces output non-zero decisions after masking by the look-up table. If there is only one decision, we retain that decision as the final decision. In the event of more than one decision, we resolve the ambiguity by retaining only the decision with the lowest classifier cost.

E. Experiment and Results

We now present results for two subjects, H.464 and H.564. It has been previously reported [2] that H.464 is a far more stable subject than H.564 and as such, its initial DP when using CSP + ECOC is much better than for H.564. In our experiment, we used data from session 1 and 2 for training, and did the validation on session 3. Using the procedure of section IV, we learned 3 subspaces per direction for session 1 and 2, for a total of 48 subspaces. Furthermore, we computed the condensed subspaces for each direction, for a total of 16 condensed for both sessions. Of the 64 subspaces, we retained only the 8 best subspaces for the classification. In order to fairly compare the confidence of the decisions among subspaces, we used the same classifier, labeled C_0, which is uses CSP and ECOC and was trained on the trials from session 1 and 2 without projections. As we can see from the table I, using the subspace projection approach we get an improvement of 12% with the less stable subject, H.564, and an improvement of 4% with more stable H.464, to achieve close to 90% DP with both subjects in the tough scenario of inter-session experiments. Ideally, with enough training sessions available, we should find intersections between subspaces from different sessions to get a truly recurrent inter-session subspace. Clearly, the intersection should be done with the condensed subspaces to reduce complexity.

TABLE I

DECODING POWER COMPARISON FOR TWO SUBJECTS

Subject	CSP+ECOC	CSP+ECOC+Subspaces
H.564	76%	88%
H.464	86%	90%

VI. CONCLUSION

We introduced a subspace approach for learning recurrent features from multi-channel LFP data, and showed its potential in movement direction decoding. We motivated the pursuit of recurrent subspaces to model the stable behavior of the sources over time. Furthermore, we proposed a classification methodology that uses multiple subspace projections while allowing us to leverage state-of-the-art classifiers to capture the spatial diversity of the mixing. We find DP improvement from 76% to 88% for subject H.564 under the challenging condition of inter-session training. We are currently investigating an approach for finding subspace intersection across sessions to improve the robustness of our method.

REFERENCES

[1] B. Vikrham Gowreesunker and Ahmed H. Tewfik, "A shift tolerant dictionary training method," in *Proc. Signal Processing with Adaptive Sparse Structusred Representations*, Saint-Malo, France, April 2009.

[2] Nuri F. Ince, Rahul Gupta, Sami Arica, Ahmed H. Tewfik, James Ashe, and Giuseppe Pellizzer, "Movement direction decoding with spatial patterns of local field potentials," in *To appear in Proc. IEEE Neural Engineering Congress*, Antalya, Turkey, 2009.

[3] Paul D. Ogrady, Barak A. Pearlmutter, and Scott T. Rickard, "Survey of sparse and non-sparse methods in source separation," *International Journal of Imaging Systems and Technology*, vol. 15, pp. 18–33, 2005.

[4] B. Vikrham Gowreesunker and Ahmed H. Tewfik, "Blind source separation using monochannel overcomplete dictionaries," in *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, May 2008.

[5] Thomas G Dietterich and Ghulum Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Artificial Intelligence Research*, vol. 2, pp. 263–286, 1995.

[6] H. Ramoser, J. Mller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial eeg during imagined hand movement," *IEEE Transaction on Rehab. Eng*, vol. 8, pp. 441–446, 2000.

[7] S. Chen, C.F. Cowan, and P.M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE transactions on neural networks*, vol. 2, no. 2, pp. 302–309, 1991.

[8] G. H. Golub and C. F. van Loan, *Matrix Computations*, Johns Hopkins University Press, 3rd edition, 1996.