NEURAL DECODING USING A NONLINEAR GENERATIVE MODEL FOR BRAIN-COMPUTER INTERFACE

Henrique Dantas[1], Spencer Kellis[2], V John Mathews[3], and Bradley Greger[4]

[1]Universidade Federal de Pernambuco, Recife-PE – Brazil
[2] Biology and Biological Engineering Division, California Institute of Technology, Pasadena, CA 91125
[3]Department of Electrical & Computer Engineering, University of Utah, Salt Lake City, UT 84109
[4] School of Biological & Health Systems Engineering, Arizona State University, Tempe, AZ 85287

ABSTRACT

Kalman filters have been used to decode neural signals and estimate hand kinematics in many studies. However, most prior work assumes a linear system model, an assumption that is almost certainly violated by neural systems. In this paper, we show that adding nonlinearities to the decoding algorithm improves the accuracy of tracking hand movements using neural signal acquired via a 32-channel micro-electrocorticographic (μECoG) grid placed over the arm and hand representations in the motor cortex. Experimental comparisons indicate that a Kalman filter with a fifth order polynomial generative model relating the hand kinematics signals to the neural signals improved the mean-square tracking performance in the hand movements over a conventional Kalman filter employing a linear system model. This finding is in accord with the current neurophysiological understanding of the decoded signals.

Index Terms— Neural decoding, Brain-Computer Interface, Nonlinear Kalman Filter

1. INTRODUCTION

Brain-Computer Interfaces (BCIs) can be used to restore movement and communication for people with tetraplegia, brain stroke, or other infirmities. One of the biggest challenges facing BCIs is the formulation of a model for predicting effector kinematics from neural activity. Due perhaps to the complexity of the underlying biomechanical system for generating movement, there are a wide variety of such models in the literature. Some of these models are nonlinear [1-4], but most are linear, and many of these linear models fall into broad categories of Wiener filters [5-6], population vector [7-8], probabilistic methods [9-11], and recursive Bayesian decoders such as Kalman filters [12-16].

The Kalman framework, which is one solution to the problem of using noisy measurements to update the state of an entity with known dynamics, is well suited to the BCI problem, which uses information extracted from neural signals to update the state of a prosthetic effector. Many BCI implementations in the literature have therefore used Kalman filters [12-16]. By incorporating prior knowledge about the effector's dynamical constraints and balancing the influence of the effector prior and neural innovations based on their relative certainty, Kalman filters offer a systematic way to deal with error. Furthermore, Kalman filters can be implemented in a computationally efficient manner, making them appropriate for real-time operation [14].

A number of improvements have been proposed to adapt Kalman filters to specific BCI design problems. Gilja et al. proposed several changes to the way in which the Kalman filter is parameterized to explicitly recognize the presence of a user interacting with the system [15]. Mulliken et al. incorporated goal information in the Kalman filter model [16]. Others have noted that the linearity assumption of the standard Kalman filter model may not optimally describe the data, and several nonlinear techniques have been explored including generalized linear models, generalized additive models, neural networks, and global Laplacian models [1-3]. Many of these models demonstrated better coverage of the data, but suffered from high computational cost. An unscented Kalman filter (UKF) which incorporates a quadratic function of neural tuning and 10th-order autoregressive modeling of the hand trajectories was proposed in [4]. This system demonstrated better signal-to-noise ratio in the outputs with the nonlinear model and kinematic memory than with a standard Kalman filter [4]. However, the system in [4] still employed a linear model to relate the neural signals to the hand movements.

In addition to model selection, BCIs may also be differentiated by the area of the brain from which the neural signals are derived (e.g., motor cortex [1-15] or parietal cortex [16]), the scope of the recorded neural signal (e.g. neuronal spikes [1-16] or field potentials [17-19]), and by the particular kinematic variables used in the model to produce effector movement (e.g. position or velocity [13,15]). The combined set of factors including neural model, brain area, neural signal, and kinematic variables represent some of the most important decisions to be made for BCI design.

This work explores offline decoding of hand trajectories from surface local field potentials recorded over hand and arm

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

area of motor cortex. Unlike the work in [4] that incorporates a quadratic state update model for hand kinematics, the approach of this paper incorporates nonlinearities of the neural signals in the generative model. A computationally efficient realization of the system was developed by subsampling the neural and hand movement data prior to implementing the Kalman filter. Analysis results presented in Section 3 will demonstrate that these modifications of the Kalman filter resulted in substantial improvement in the tracking capability of the BCI system.

2. A NONLINEAR KALMAN FILTER FOR BCI

Let $\boldsymbol{z}_k \in \mathbb{R}^C$ represents a C-element vector containing the neural signals acquired at time t_k from C different neural channels. Let $\mathbf{Z}_k = [\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \mathbf{z}_k]^T$ represent the data matrix containing the neural signals acquired till time k. Similarly, let $\boldsymbol{x}_k \in \mathbb{R}^{4p}$ be a $4p$-element vector containing the most recent p instances of the hand position (x and y locations) and hand movement velocity (x and y directions) at time k and let $\mathbf{X}_k = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k]^T$ be the data matrix containing the hand kinematic state history from time 1 through k.

The linear generative model that relates the neural signal $\mathbf{z}_k$ to the state vector $\mathbf{x}_k$ is given by

$$\boldsymbol{z}_k = \mathbf{H}\boldsymbol{x}_k + \boldsymbol{q}_k \tag{1}$$

where $\boldsymbol{q}_k$ is the modeling error vector that is assumed to be Gaussian with zero mean value and covariance matrix $\mathbf{Q}$ and $\mathbf{H}$ is the coefficient matrix that relates the state vector to the neural data, respectively.

We assume that the temporal evolution of the state vector is linear, and is given by

$$\boldsymbol{x}_{k+1} = \mathbf{A}\boldsymbol{x}_k + \boldsymbol{w}_k \tag{2}$$

where $\boldsymbol{w}_k$ is the prediction error vector that is also assumed to be Gaussian with zero mean value and covariance matrix $\mathbf{W}$, and $\mathbf{A}$ is the predictor matrix for the state vector. We note that the above equation represents a p^{th} order autoregressive model for the evolution of the state vector.

Despite assumptions of linearity in most BCIs, linear relationships are often insufficient to describe neural tuning models [4]. Linear models have worked well since they are usually more computationally efficient in implementation, and they may be more stable with nonstationary inputs. However, nonlinear models may describe neural systems better and therefore produce more accurate control of the effector [20,21]. We propose adding nonlinearities to the system in such a way as to preserve the benefits of the linear Kalman filter while providing better control. We do this by augmenting the neural signal vector to include higher order powers of the signals as

$$\boldsymbol{z}_k = [z_{k,1}, z_{k,1}^2 - E\{z_{k,1}^2\}, \ldots, z_{k,1}^m - E\{z_{k,1}^m\}, \ldots, z_{k,C}^m - E\{z_{k,C}^m\}] \tag{3}$$

Table 1: Kalman filter update equations.

A priori state update:	$\hat{\mathbf{x}}_k^- = \mathbf{A}\hat{\mathbf{x}}_{k-1}$
A priori error covariance matrix:	$\boldsymbol{P}_k^- = \boldsymbol{A}\boldsymbol{P}_{k-1}\boldsymbol{A}^T + \boldsymbol{W}$
Kalman gain:	$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}^T(\mathbf{H}\mathbf{P}_k^-\mathbf{H}^T + \mathbf{Q})^{-1}$
State update:	$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_k^-)$
Error covariance matrix:	$\mathbf{P}_k = (\mathrm{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_k^-$

where $E\{.\}$ denotes the statistical expectation of the variable within the parenthesis. We have assumed above that the neural signals have zero mean value, or that the mean values have been removed from the neural signals also. The mean values of the higher-order terms can be calculated using temporal averaging and removed to create the augmented neural signal vector.

Assuming knowledge of the coefficient matrices $\mathbf{H}$ and $\mathbf{A}$, and the covariance matrices $\mathbf{Q}$ and $\mathbf{W}$, we can implement the Kalman filter for the above model as shown in Table 1 [22].

2.1. Training the Kalman filter

Given a training sequence of neural signals and the corresponding hand locations and velocities, the matrixes A and H may be easily calculated using a least-squares approach. Assuming data spanning time samples from 1 through M, the estimates are given by

$$\widehat{\mathbf{A}} = (\mathbf{X}_{M-1}^{\mathrm{T}}\mathbf{X}_{M-1})^{-1}\,\mathbf{X}_{M-1}^{\mathrm{T}}\mathbf{X}_M \tag{4}$$

and

$$\widehat{\mathbf{H}} = (\mathbf{X}_M^{\mathrm{T}}\mathbf{X}_M)^{-1}\,\mathbf{X}_M^{\mathrm{T}}\mathbf{Z}_M \tag{5}$$

where the hats denote estimated values, and an appropriate number of zeros have been padded to $\mathbf{X}_{M-1}$ to make it compatible for multiplication with $\mathbf{X}_M$ in (4). After these coefficient matrices are estimated, we can estimate the covariance matrices from the training set as

$$\widehat{\boldsymbol{W}} = \frac{1}{M-1}\left(\boldsymbol{X}_M - \boldsymbol{X}_{M-1}\widehat{\mathbf{A}}\right)^T(\boldsymbol{X}_M - \boldsymbol{X}_{M-1}\widehat{\mathbf{A}}) \tag{6}$$

and

$$\widehat{\mathbf{Q}} = \frac{1}{\mathrm{M}}(\mathbf{Z}_M - \mathbf{X}_M\widehat{\mathbf{H}})^{\mathrm{T}}(\mathbf{Z}_M - \mathbf{X}_M\widehat{\mathbf{H}}) \tag{7}$$

In practical realizations of the Kalman filter in BCI applications, we will first train the system to estimate the parameters that cannot be updated online as shown in (4)-(7), and then implement the system in Table 1 with the estimated parameters substituting for their true values in the table.

2.2. Channel Selection

It is thought that cortical columns approximately several hundred microns in diameter form the basic computational unit

of the cerebral cortex. We, therefore, suspected that there may be redundant information in neural signals sampled at a spatial frequency of 1mm – not all channels were needed to perform an accurate decoding. Mutual information analysis was used to investigate this idea.

The mutual information between a neural signal $\mathbf{z}_k$ and the hand signals $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as

$$I_k(\mathbf{z}_k, \boldsymbol{x}, \boldsymbol{y}) = \iiint f_{z,x,y}(z,x,y) \log \frac{f_{z,x,y}(z,x,y)}{f_z(z) f_x(x) f_y(y)} dz dx dy \quad (8)$$

where the functions *f* represent the probability density functions of the subscripted variables. Ideally, to pick *N* different neural channels, we should compute the joint mutual information of all possible combinations of *N* channels with the hand signals, and pick the subset of neural channels with the highest mutual information with the hand signals.

Because of the computational complexity and the large data requirement associated with estimating the joint mutual information of a large number of variables, we used a sub-optimal approach in which the joint mutual information of signals from three neural channels and the two hand signals were first estimated for all possible triplets of neural channels. These triplets of neural channels were arranged in descending order of the corresponding mutual information values, and the first *N* channels that appeared in the ordered list was employed in the Kalman filter. Even though sub-optimal because the method does not take into account the possible redundancies between the individual channels, this approach resulted in the selection of a subset of the channels that gave acceptable performance.

3. RESULTS

3.1. Data Acquisition

The data used in this paper were collected from a patient undergoing ECoG recordings prior to the surgical treatment of epilepsy [17-19]. The patient was implanted with two 16-channel nonpenetrating micro-wire arrays that consisted of a 40 micron wire embedded in a silicone substrate with 1 mm interelectrode spacing (PMT Corporation, Chanhassen, MN). The arrays were placed in the epipial space underneath a standard clinical electrocorticographic (ECoG) grid. One of the arrays was placed over arm and hand representations in motor cortex as confirmed with intraoperative somatosensory evoked potential (SSEP) monitoring. Both arrays were referenced to low-impedance wires placed in the epidural space. Thirty-two channels of neural data and two channels with X and Y hand position data recorded using a sampling rate of 30 kHz.

The patient performed a simple reaching task using a mouse on a draftsman tablet (20 cm x 20 cm). The patient moved the mouse from the bottom center of the table to the upper left or upper right corner of the table. In addition to the position data, we also calculated the velocity as the first order derivative of the position signals.

Since hand movements are low frequency signals (it is practically difficult to move the hands at rates faster than 4 Hz.), our system preprocessed both the hand and neural signals with a lowpass filter with 10 Hz. cut off frequency and then sub-sampled the resulting signals to implement a computationally efficient update strategy for the nonlinear Kalman filter at the rate of 60 samples/s.

3.2 Performance Evaluation

For all the analysis presented in this paper, the data were broken into seven parts. The Kalman filter was trained on the first part, and then tested on the other six. During the training process, the coefficient matrices **A** and **H**, along with the covariance matrices **W** and **Q** were computed as in (4)-(7). The remainder of the calculations were made as in the Kalman filter iterations of Table 1 even for the training block. For the other six blocks, the matrices **A**, **H**, **W** and **Q** were kept frozen as estimated from the training block and the remainder of the iterations were performed as in Table 1. In most cases, the results shown here are averages computed over blocks 2 through 7 used as the test data.

The first set of analysis was performed to evaluate the best delay to introduce into the neural signals before they are fed to the Kalman filter. For this, we used a linear generative model as well as a nonlinear 5th order generative model along with a 5th order autoregressive state update model, and implemented a generative model of the form

$$\boldsymbol{z}_{k-r} = \mathbf{H}\boldsymbol{x}_k + \boldsymbol{q}_k \quad (9)$$

and estimated the mean-square estimation error for the hand movement signal for different values of the delay value *r*. The results are plotted in Figure 1. They were obtained using all 32 neural channels. The results indicate that (1) the nonlinear generative model outperforms the linear model; and (2) there is an "optimum" value of uniform delay (all channels are delayed

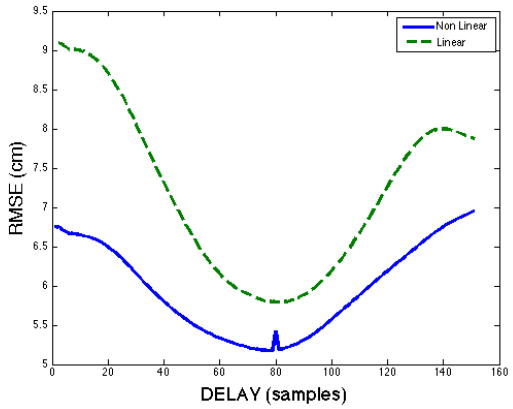

Fig. 1: Root-mean-square estimation error as a function of the delay in the neural signal

by the same number of samples) for which the Kalman filter performs the best in the least-squares sense. Both these observations are in accord with our current neurophysiological understanding of the decoded signals.

The optimal delay observed in the above results is somewhat higher than the typical delay values of a few hundred milliseconds between neural signals and the hand signals reported in the literature. This variation from prior results may be due to several reasons. First, the measurements were performed soon after the implantation prior to complete healing of the injuries from the surgery. This as well as the effects of medication may have slowed the response time. A fraction of the observed delay may also be caused by latencies in the data acquisition hardware.

Figure 2 displays the effect of adding nonlinearities in the generative model in the Kalman filter. Plotted here are the correlation coefficients between the estimated hand movements and the actual measurements of the hand movements as a function of the order of nonlinearity for an autoregressive state update model and an autoregressive state update model of order 5 for nonlinearity orders 1 through 5. We can see from the results that the nonlinearities in the model improves the quality of estimates substantially. Furthermore, additional memory in the autoregressive model improves the performance of the system. However, this improvement is less for higher-order nonlinearities.

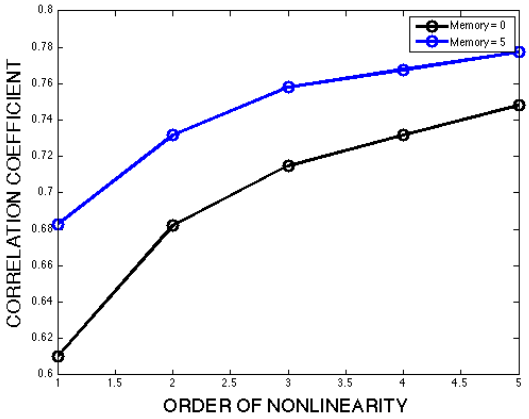

Fig. 2: Correlation coefficient of estimated and measured hand position signals as a function of the order of nonlinearity in the generative model. Two cases, corresponding to a first-order autoregressive model for state update and an update model of order 5 are shown.

In the last set of analysis, we selected 15 and 10 channels based on mutual information analysis to estimate the hand movements. The correlation coefficients of the estimated hand movements with the measured values for 32, 15 and 10 channels, a 5th order nonlinear generative model, and a 5th order autoregressive state update model is shown in Figure 3.

Plots of the hand movements (only in *x*-direction) obtained using 32, 15 and 10 neural channels are shown in Figure 4. It appears from the results in Figures 3 and 4 that our approach to selecting the neural channels using sub-optimal mutual information analysis is able to identify the channels that contribute most to the estimation of hand movements. In particular, there is little loss of information when the number of channels is reduced to 15 from the original 32.

4. CONCLUDING REMARKS

This study shows that the performance of brain-computer interface systems for tracking hand movements using neural signals acquired via micro-electrocorticographic grids can be substantially improved if nonlinearities involving neural signals are added to the generative model. Furthermore, at 1 mm spacing of the electrode, there is sufficient redundancy in the neural signals that not all neural channels are needed to perform efficient kinematic hand decoding from brain signals. Additional research involving optimal selection of the nonlinear models, computationally efficient, mutual information-based, optimal channel selection, channel-specific delay estimation and algorithms to blindly update the "frozen" parameters of the system are currently underway.

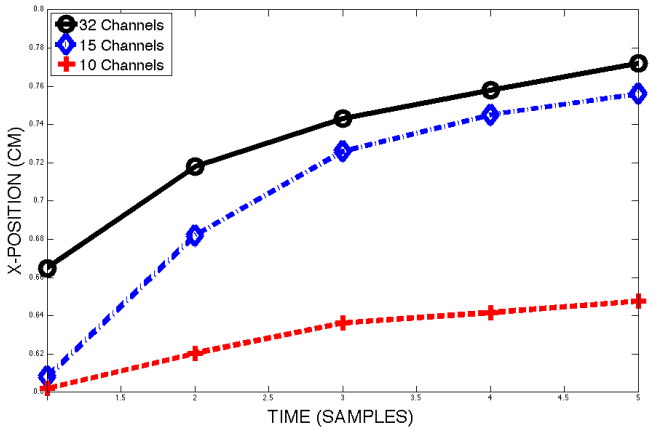

Figure 3. Correlation coefficients of the estimated hand movements for reduced number of neural channels.

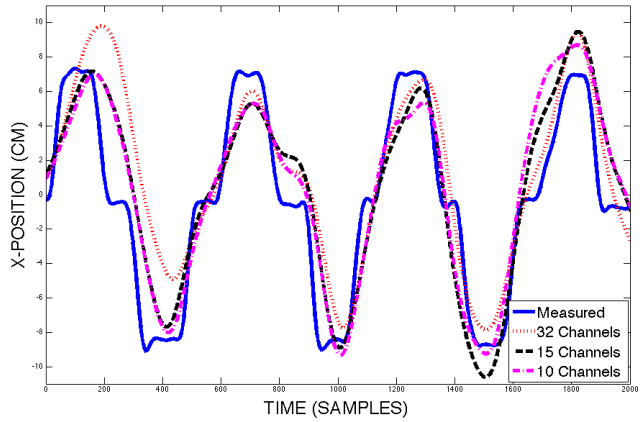

Figure 4. Comparison of the estimated hand movements using nonlinear models with 32, 15 and 10 channels. Only movements along the *x*-direction are shown.

REFERENCES

1. Y. Gao, M. J. Black, E. Bienenstock, W. Wu, J. P. Donoghue, "A quantitative comparison of linear and non-linear models of motor cortical activity for the encoding and decoding of arm motions," *Neural Engineering, Conf. Proc. IEEE EMBS*, pp. 189-192, March 2003.
2. S-P Kim, J. C. Sanchez, Y. N. Rao, D. Erdogmus, J. M. Carmena, M. A. Lebedev, M. A. L. Nicolelis, J. C. Principe, "A comparison of optimal MIMO linear and nonlinear models for brain-machine interfaces," *Journal of Neural Eng*, vol. 3, pp. 145-161, 2006.
3. B. M. Yu, J. P. Cunningham, K. V. Shenoy, M. Sahani, "Neural decoding of movements: From linear to nonlinear trajectory models," *Neural Information Processing: Lecture Notes in Computer Science*, vol. 4984, pp. 586-595, 2008.
4. Z. Li, J. O'Doherty, T. Hanson, M. Lebedev, C. Henriquez, M. Nicolelis, "Unscented Kalman Filter for Brain-Machine Interfaces," PLoS ONE, vol. 4, no. 7, 2009.
5. J. Wessberg, C. R. Stambaugh, et. al, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature*, vol. 408, pp. 361-365, 2000.
6. L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn, and J. P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature*, vol. 442, no. 7099, pp. 164–171, Jul. 2006.
7. A. P. Georgopoulos, R. E. Kettner, A. B. Schwartz, "Primate motor cortex and free arm movements to visual targets in three-dimensional space. II. Coding of the direction of movement by a neuronal population," *J Neurosci*, vol. 8, pp. 2928-2937, 1988.
8. D. M. Taylor, S. I. Helms Tillery, A. B. Schwartz, "Information conveyed through brain-control: Cursor versus robot," *IEEE Trans Neural Sys Rehab*, vol. 11, no. 2, Jun. 2003.
9. W. Wu, M. J. Black, D. Mumford, Y. Gao, E. Bienenstock, J. P. Donoghue, "Modeling and decoding motor cortical activity using a switching Kalman filter," *IEEE Trans Biomed Eng*, vol. 51, no. 6, Jun. 2004.
10. M. J. Black and J. P. Donoghue, "Probabilistically modeling and decoding neural population activity in motor cortex," in *Toward Brain-Computer Interfacing,* G. Dornhege, J. del R. Millan, T. Hinterberger, D. McFarland, K.-R. Muller (eds.), MIT Press, pp. 147-159, 2007.
11. Y. Gao, M. J. Black, E. Bienenstock, S. Shoham, J. P. Donoghue, "Probabilistic inference of hand motion from neural activity in motor cortex," *Proc. Adv. in Neural Info. Processing Systems* 14, The MIT Press, 2002.
12. W. Wu, Y. Gao, E. Bienenstock, J.P. Donoghue, and M. J. Black, "Bayesian population decoding of motor cortical activity using Kalman filter," *Neural Comput*., vol. 18, pp. 80-118, 2006
13. S-P Kim, J. D. Simeral, L. R. Hochberg, J. P. Donoghue, M. J. Black, "Neural control of computer cursor velocity by decoding motor cortical spiking activity in humans with tetraplegia," *J Neural Eng* vol. 5 2008.
14. W. Q. Malik, W. Truccolo, E.N.Brown and L.R Hochberg, "Efficient Decoding With Steady-State Kalman Filter in Neural Interface Systems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering,* vol. 19, no. 1, February 2011.
15. V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, M. M. Churchland, M. T. Kaufman, J. C. Kao, S. I. Ryu, K. V. Shenoy, "A high-performance neural prosthesis enabled by control algorithm design," Nature Neuroscience, vol. 15, no. 12, pp. 1752-1757, 2012.
16. G. H. Mulliken, S. Musallam, R. A. Andersen, "Decoding trajectories from posterior parietal cortex ensembles," *J Neurosci*, vol. 28, pp. 12913-12926, 2008.
17. S. Kellis, P. House, K. Thomson, R. Brown, B. Greger, "Human neocortical electrical activity recorded on nonpenetrating microwire arrays: applicability for neuroprostheses," *Neurosurg Focus*, vol. 27, p. E9, Jul 2009.
18. S. Kellis, K. Miller, K. Thomson, R. Brown, P. House, B. Greger, "Decoding spoken words using local field potentials recorded from the cortical surface," *J Neural Eng* vol. 7, 2010.
19. S Kellis, S. Hanrahan, T. Davis, P. A. House, R. Brown, B. Greger, "Decoding hand trajectories from micro-electrocorticography in human patients," *Conf Proc IEEE Eng Med Biol Soc*, vol. 1, pp. 4091-4, 2012.
20. B. A. Olshausen and D. J. Field, "How close are we to understanding V1?" *Neural Computation*, vol. 17, pp. 1665-1669, 2005.
21. N. C. Rust, O. Schwartz, J. A. Movshon and E. P. Simoncelli, "Spatitemporal elements of macaque V1 receptive fields," *Neuron*, vol. 46, pp. 945-956, June 2005.
22. A. V. Balakrishnan, *Kalman Filtering Theory*, Springer-Verlag, Secaucus, New Jersey, 1984.