Neurally Constrained Subspace Learning of Reach and Grasp

Payman Yadollahpour*, Greg Shakhnarovich†, Carlos Vargas-Irwin‡, John Donoghue‡, Michael Black*
*Department of Computer Science, Brown University, Providence, RI 02912, USA
Email: {pyadolla, black}@cs.brown.edu
†Toyota Technological Institute, Chicago, IL 60637, USA
Email: greg@tti-c.org
‡Department of Neuroscience, Brown University, Providence, RI 02912, USA
Email: {carlos_vargas_irwin, john_donoghue}@brown.edu

***Abstract*—Neural decoding of motor control of hand and arm movements in primates is a challenging task that requires developing statistical models that explain how the recorded neural population activity relates to motor behavior. Until recently, much of the work in this area has focused on learning linear models of decoding for low-dimensional motor control, such as 2D control of a computer cursor. Capturing a richer set of motor behaviors such as hand and arm posture during object grasping and manipulation tasks introduces much higher dimensional representations of motor control. Understanding the underlying degrees of freedom in complex kinematics that are explained by the neural activity is a central question. One way of learning these "effective" degrees of freedom has been to employ dimensionality reduction techniques, such as Principal Component Analysis, to find a linear kinematic subspace that accounts for the observed motor behavior, separate from the observed neural activity. The orthonormal basis vectors that span this subspace are then considered as the underlying latent variables, or "motor primitives" that describe behavior. These motor primitives are not guaranteed to be optimally correlated with the observed neural activity however. In this paper we devise an objective function and optimize it to learn a linear subspace of the motor activity that tries to maximize the correlation between the latent variables of this subspace and the neural activity, while still explaining the motor behavior with reasonable fidelity.**

I. Introduction

The decoding of neural signals for complex kinematic control is a key problem that scientists and engineers face when devising new neural interface systems and prosthetic devices to help people with paralysis physically interact with their environment. Until recently much of the work has focused on decoding of neural population spiking activity from primary motor cortex for two-dimensional kinematic control [1]. These decoding methods often assume a linear relationship between neural firing rates and intended movement which may correspond to the position or velocity of a computer cursor. Our work focuses on accurately decoding a high dimensional representation of hand and arm posture using neural activity from a limited number of cells from the primary motor cortex. Two key questions arise in this problem. First we would like to know how the neural signals relate to intended posture (the encoding problem). Second we would like to enable a feasible engineering approach to building neural decoding systems that are reliable and require minimal training data.

To address these problems, previous research has examined low-dimensional linear representations of various forms. For example, Wu et al [2] used principal component analysis (PCA) to reduce the dimensionality of neural firing rate data and then modeled a linear relationship between the lower-dimensional PCA space and hand kinematics. This PCA representation removed redundant degrees of freedom in the population firing rates and thus enabled the training of the decoding algorithm with less data. Previous researchers have also explored low-dimensional representations of the kinematic state with the goal of uncovering motor "primitives" that are related to neural firing activity [3][4][5][6].

In both of these approaches, previous work has looked at either firing activity or kinematics separately. We argue that a better approach seeks low dimensional models that take both into account simultaneously. Seeking a PCA representation of the kinematics alone may represent irrelevant degrees of freedom not accounted for by the neural activity. Instead we explicitly seek the low dimensional model of the behavior that is good for decoding. We achieve this by introducing a modified optimization objective that reflects both constraints.

II. Previous Work

Dimensionality reduction methods have been used extensively in studies of motor behavior. Typically, PCA is applied in the space of postures, represented in terms of positions [5] or joint angles [6]. These studies have been supported by others that have looked at the independence relationships between fingers of the hand. Previous studies have shown that the independence in the fingers of the hand is limited due to mechanical coupling and active neuromusculur control [7], and that the effective degrees-of-freedom is much fewer than the theoretically available degrees-of-freedom [8]. In fact, [9][4] showed that due to the correlations between finger joints in the hand, that applying PCA on the joint angle space of the hand produced two to three principal components that accounted for most of the variance in the hand posture. Such analysis provides insights into variability of posture.

More recently, PCA has been used on dynamic movement data. Technically this is achieved by constructing time series of fixed length; the simplest way consists of taking contiguous sub-sequences of frames, at regularly spaced points in time. A subspace in such a space represents both spatial and temporal properties of the underlying signal. For example, in [2] PCA was applied on the neural data (stacked history of firing rates). The coefficients of the resulting expansion were used as the input to Kalman filter, decoding hand position and velocity in a variety of two-dimensional tasks.

In a recent work [3] closely related to our paper, time series subject to dimensionality reduction are sequences of hand velocity estimates. The goal is to model the distribution of spiking activity for a set of cells as a function of the underlying movement. After the principal components are learned, a log-linear preferred direction tuning model is fit in the principal subspace.

In much of the work mentioned above, there are two related data streams: neural activity represented by firing rates, and the corresponding motor behavior represented by the measured kinematics. However, dimensionality reduction is carried out on one of the spaces (either neural or kinematic), without explicit regard to the underlying goals of decoding/encoding the other space. The role of PCA in these cases was limited to reducing the dimension for computational expediency, decorrelating the kinematic data, and potentially reducing noise.

In contrast, the idea of directly optimizing a lower-dimensional representation to account for the *relationship* between two spaces has been proposed in other fields. Canonical Correlation Analysis (CCA) is a well known example that learns a pair of linear subspaces that maximize the correlation between projections of the data sets onto these subspaces. Partial Least Squares (PLS) extends CCA by trying to balance the criterion of maximizing correlation between the projections and maximizing the variance in the two data-sets that is explained by the learned subspaces. In other work [10] linear subspaces for two spaces are learned with the objective to optimize a mapping from one space to the other.

Our work combines the experimental framework of [3] and the idea of coupled subspace learning from [10].

III. Kinematic Data

The kinematic data consists of temporally changing joint angle measurements from all of the segments in the hand and arm of two rhesus macaque monkeys. Optical motion capture technology was used to capture the posture of the hand and arm segments in 3D space. To do this 12 optical motion capture cameras were used to triangulate the position of 4mm reflective optical markers that were placed on each segment of the hand and arm, starting at the shoulder and extending to the fingers of the hand. The recorded 3D positions of the segments at each time frame in the recordings was then fit in a least squares fashion to a kinematic model of the hand and arm. The fit kinematic model was then used to determine Euler angle measurements for the rotation of each segment relative to its parent. The proximal and distal interphalangeal (PIP and DIP) joints of the fingers as well as the elbow joint were constrained to be one degree-of-freedom (DoF) hinge joints, the metacarpophalangeal (MCP) joints as well as the palm were constrained to be 2-DoF joints, and the shoulder was represented as a 3-DoF ball-and-socket joint. To remove singularities present in the Euler representation of the shoulder, and maintain consistent representation for segment rotations, we converted the Euler angles to exponential map and twist representation. The kinematic data that was collected consisted of motion capture recordings of hand and arm activity in reaching and grasping tasks performed by two monkeys. The data was collected over four sessions on separate days, each session consisting of six to nine trials, where each trial was comprised of twenty to thirty reach and grasp tasks of various objects that were presented to the monkeys.

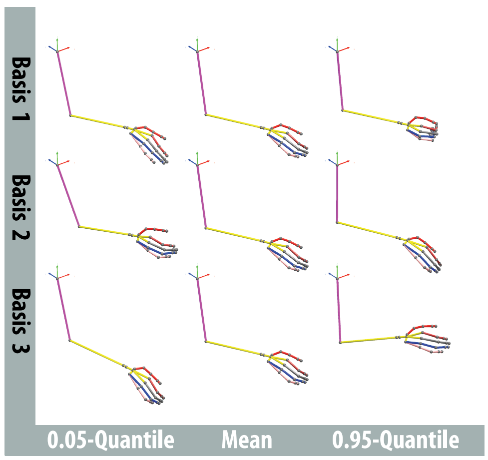
A. Exponential Map and Twist Representation

Euler angle representations of 3-DoF rotations suffer from a commonly known issue in the graphics community known as gimbal lock. To overcome this limitation we converted the computed Euler angles of the fit kinematic model to a three element vector exponential map and twist representation [11]. An example of the conversion in rotational representation can be given for the shoulder joint. The 3-DoF rotation of the shoulder joint can be separated into a twist about the major axis of the upper arm segment in its canonical position (hand stretched out to the side) and a swing of the upper arm. In order to ensure that the swing component of the rotation doesn't have a rotational component about the twist axis we need to choose a rotational axis for the swing that is orthogonal to the vector pointing in the canonical direction of the upper arm. This means that the exponential map rotation vector for the swing component has to lie on a plane perpendicular to major axis of the upper arm in its canonical pose. Since the canonical pose of the upper arm is stretched out to the side, choosing the plane to be the sagittal plane ensures separate twist and swing components, and prevents the singularity that occurs at $\pm\pi$. The singularity that occurs when the twist angle is $\pm\pi$ is prevented due to the physical constraint of positioning the arm passed the upper torso. The three element exponential map/twist vector can be constructed as follows,

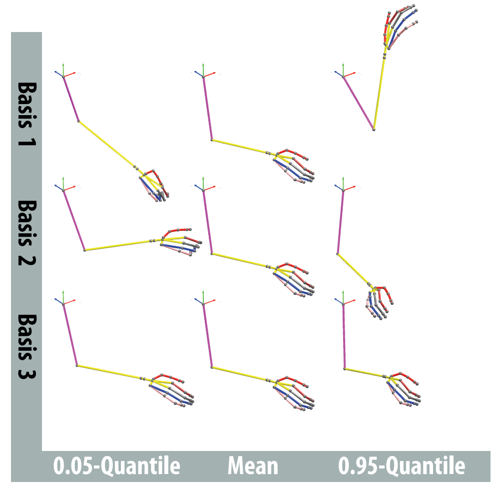
$$\mathbf{e} = [e_1, e_2, e_3] \tag{1}$$

where $\mathbf{v} = [e_1, e_2]$ is the swing component and e_3 is a 1-DoF Euler angle twist angle. The compact exponential map representation for the rotation angle and unit axis of rotation for the swing are $\theta = |\mathbf{v}|$ and $\hat{\mathbf{v}} = \frac{\mathbf{v}}{|\mathbf{v}|}$ respectively. To compute the rotational axis for the swing component, $\hat{\mathbf{v}}$, as well as θ and e_3, we first applied the previously computed Euler angle rotations, from the fit of the kinematic model to the observed markers, to a 3×3 matrix $\mathbf{P}$, representing a coordinate frame that was oriented along the major axis of the segment in its canonical pose. This resulted in a new coordinate matrix $\mathbf{P}'$ aligned to new pose of the segment defined by the Euler angles. The coordinate matrices can be written as,

$$\mathbf{P} = [\mathbf{l}^T, \mathbf{u}^T, \mathbf{n}^T] \tag{2}$$



(a) PCA subspace.



(b) Optimal learned subspace.

Fig. 1. (**a**) The first three principal components of posture for a single recorded session. The middle column shows mean posture. Left and right columns show the posture at the two extremes of the data projected onto the principal component direction. (**b**) First three basis vectors spanning optimal learned subspace.

where $\mathbf{l}$, $\mathbf{u}$, $\mathbf{v}$, are unit vectors pointing along the major axis, and the two orthogonal directions respectively. Similarly,

$$\mathbf{P}' = [\mathbf{l}'^T, \mathbf{u}'^T, \mathbf{n}'^T] \tag{3}$$

The unit rotational axis for the twist component was then computed as follows,

$$\hat{\mathbf{v}} = \mathbf{l} \times \mathbf{l}' \tag{4}$$

The magnitude of rotation becomes,

$$\theta = \arccos(\mathbf{l} \cdot \mathbf{l}') \tag{5}$$

To find the twist rotation angle, e_3, we first align $\mathbf{l}$ and $\mathbf{l}'$ by applying a rotation of $-\theta$ about $\hat{\mathbf{v}}$ to $\mathbf{P}'$ and then compute the inner product between $\mathbf{u}$ and $\mathbf{u}'$ as follows,

$$e_3 = \arccos(\mathbf{u} \cdot \mathbf{u}') \tag{6}$$

The three element exponential map/twist representation of all the other 2-DoF hardyspicer joints in the arm and hand were computed in a similar way. The rotational axis for the twist vector was always chosen to be aligned with the major axis of each segment in its canonical pose and the plane of the rotational axis for the twist component was always chosen to be orthogonal to the major axis. For each of the 1-DoF hinge joints a single variable was used to represent the extension/flexion angle of the joint. This resulted in a representation of the kinematic pose consisting of 32 degrees-of-freedom.

IV. Dimensionality Reduction - PCA

A key insight in neural decoding of motor control is to remove redundant representations of data in either the motor activity or neural activity. Redundant representation makes the decoding task more computationally expensive and provides no extra information to help us learn the decoding model. Therefore we want to only use the "effective" degrees of freedom present in the data. The standard method for removing the redundant degrees of freedom in a data set is by employing a dimensionality reduction technique such as Principal Component Analysis (PCA). PCA makes use of the first and second order statistics (ie. mean and covariance) of the data to compute a set of orthonormal principal components, or basis vectors, such that the projection of the data onto the basis vectors are maximally uncorrelated. The projections of the data onto the basis vectors can be thought of as random variables. The original data is then a linearly weighted combination of these random variables. If the joint distribution of the random variables is Gaussian then the random variables are statistically independent. Dimensionality reduction can be applied to either the neural firing rates or the motor activity or both. When applying PCA, or other dimensionality reduction methods, to a data set of recorded motor activity these random variables can be thought of as motor "primitives". This paper focuses on applying dimensionality reduction techniques on the motor activity.

As a first step towards learning the motor primitives Principal Component Analysis was applied to the space of hand and arm postures represented by 60 percent of the recorded kinematic data. PCA was applied to each of the four recorded reaching and grasping sessions separately. The percent variance plot as a function of the number of principal components (basis vectors) is shown in Fig. 2.

From Fig. 2 constructing a basis consisting of the first six eigenvectors (principal components) can explain roughly 90 percent of the variance in the hand and arm posture, and yet provide a much lower dimensional representation of the kinematic space than the full 32-DoF space. This suggests that there is a great deal of correlation between the 32-DoF. These

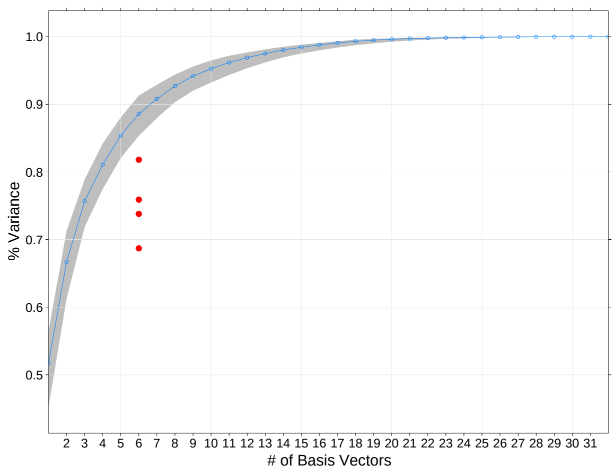


Fig. 2. Percent variance covered in the kinematic data as a function of the number of principal components (basis vectors). Basis vectors ordered in decreasing order of eignenvalues. Solid blue line indicates mean values and the shaded region covers the range of values over the four recorded sessions. The red dots indicate percent cumulative variance accounted for by the learned subspace $\mathbf{U}$ for the four sessions.

correlations between the angular degrees-of-freedom in the hand and arm introduce unnecessary redundancy in the representation of hand and arm posture and the redundant degrees-of-freedom can be removed without loss of information. Of course in reducing the kinematic posture representation from 32-DoF to 6-DoF we are balancing between capturing the effective degrees of freedom in the kinematics and providing a much sparser representation for hand and arm posture. The first three of these six principal components for the training data in one of the four recorded kinematic sessions is shown in Fig. 1a.

From Fig. 1a we see that the first basis vector accounts for most of the variance found in the joints of the hand, whereas basis vectors two and three account for the variance found in the forearm rotation and elbow angle respectively. Though applying a dimensionality reduction method, such as PCA, to the data set of recorded motor activity gives some insight into the underlying motor primitives that might be controlled by the motor cortex there is little to justify that these basis vectors are the motor primitives that the brain codes for. This is because such methods do not take into account correlation of the projection coefficients (ie. random variables) with neural activity. Rather, what we want is to incorporate the knowledge of neural activity into learning a set of basis vectors such that the projected kinematic data onto these basis vectors can be explained by the observed neural firing rates.

V. Neurally Coupled Subspace Learning

Given recordings of the neural firing rates of a population of motor cortical neurons and the relative orientations of all the segments in the hand and arm, the goal is to learn a low-dimensional latent space representation of the hand and arm posture. Projection of the kinematic data onto these basis vectors provides a set of random variables representing the contribution of each basis vector on the hand and arm posture. We suggest that decoding these few random variables as opposed to the full set of 32 degrees-of-freedom will provide for more efficient decoding algorithms, require less data to train the decoding model, and still maintain good hand and arm posture reconstruction accuracy.

We will term the space of measured segment orientations for all the segments in the hand and arm the *full kinematic space*. The segment orientations are defined at each recorded time instant as 3 element exponential map and twist vector rotations about the joint of each segment's parent. The skeletal hierarchy originates at the shoulder and extends to the tips of the fingers observing the physiological linkages at the joints where parent and child segments meet. The measured rotations at a joint are relative to the parent segment's orientation, and the shoulder segment orientation is measured at the shoulder joint relative to the global coordinate system.

Given the full kinematic space representation of the hand and arm posture at each time instant we propose that there is a linear subspace representation of the full kinematic space that captures the important degrees of freedom present in the full kinematic space such that the recorded neural firing rates are linearly correlated with the variables lying in the subspace. The d-dimensional full kinematic space can be represented as a matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$ where N is the number of observations, and at each time instant t we observe the kinematics $\mathbf{x}_t = [x^t_{i=1}, \ldots, x^t_{i=d}]^T$. Given the full kinematic space $\mathbf{X}$ the task is to learn a p-dimensional linear subspace $\mathbf{U}$ spanned by a set of orthonormal basis vectors, $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_p]$. The basis $\mathbf{U}$ has dimensionality $d \times p$ such that $p < rank(\mathbf{X}) <= d$. The p latent variables at time t, $\mathbf{c}_t = [c^t_{j=1}, \ldots, c^t_{j=p}]^T$, are computed by projection of $\mathbf{x}_t$ onto the subspace defined by $\mathbf{U}$. This can be written as $\mathbf{C} = \mathbf{U}^T\mathbf{X}$, where $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_N]$.

Given the subspace $\mathbf{U}$ we assume that we can learn a linear decoding model relating the neural population firing rates $\mathbf{Z}$ to the latent variables $\mathbf{C}$. Given the history of firing rates of a population of k neurons in the past h time instances we construct the firing rate history matrix as $\mathbf{Z} = [\mathbf{z}(1)^T_0, \ldots, \mathbf{z}(1)^T_h, \ldots, \mathbf{z}(k)^T_0, \ldots, \mathbf{z}(k)^T_h]^T$, where each row vector $\mathbf{z}(i)_j = [z^i_{t=0-h+j}, \ldots, z^i_{t=N-h+j}]$. The linear decoding model relating the neural firing rates to the latent variables in the kinematic subspace can then be written as,

$$\mathbf{U^T X} = \mathbf{FZ} \tag{7}$$

where $\mathbf{F}$ is the $p \times kh$ matrix of linear filter coefficients, the linear least squares solution of which can be derived by applying the Moore-Penrose pseudo-inverse of $\mathbf{Z}$ denoted as $\mathbf{Z}^\dagger$,

$$\mathbf{F} = \mathbf{U^T X Z^\dagger} \tag{8}$$

Given this linear decoding model we want to learn $\mathbf{U}$ such that the error between the actual latent variables $\mathbf{C} = \mathbf{U}^T\mathbf{X}$ and their linear prediction $\tilde{\mathbf{C}} = \mathbf{FZ}$ is minimized. In other words we want to find a subspace $\mathbf{U}$ such that

$\| \, (\mathbf{C} - \tilde{\mathbf{C}}) \, \|_2^2 = \mathbf{0}$. This gives the following function for minimizing the decoding error,

$$\arg\min_{U} \| \, (\mathbf{U^T X} - \mathbf{FZ}) \, \|_2^2 \tag{9}$$

where $\| \cdot \|_2$ is the 2-norm. The solution for $\mathbf{U}$ given in eqn. 9 will not necessarily be optimal in terms of reconstruction accuracy of the full kinematic space – that is the subspace $\mathbf{U}$ might not span the relative degrees of freedom present in the kinematic data . Another way to see this is that the approximation of the full kinematic state we get by projecting the kinematic data from the full space to the subspace and then embedding back into the original space, by way of $\mathbf{UU}^T\mathbf{X}$, might not give an accurate estimate of $\mathbf{X}$ since $p < rank(\mathbf{X})$. In order to enforce good reconstruction of the full kinematic space we introduce the following equation for minimizing the reconstruction error,

$$\arg\min_{U} \| \, (\mathbf{X} - \mathbf{UU^T X}) \, \|_2^2 \tag{10}$$

Combining the terms from eqns. 9 & 10 we aim to minimize the following objective function,

$$\zeta = \arg\min_{U} \; (1-\alpha) \, \| \, (\mathbf{X} - \mathbf{UU^T X}) \, \|_2^2 + \alpha \, \| \, (\mathbf{U^T X} - \mathbf{FZ}) \, \|_2^2 \tag{11}$$

where the constant parameter α controls the relative weight placed on the reconstruction and decoding error terms. If $\alpha = 0$ then the objective in eqn. 11 reduces to performing PCA on $\mathbf{X}$. In order to minimize eqn. 11 in terms of $\mathbf{U}$ we employ a recursive linear optimization algorithm shown below.

```
input  : PCA basis U_0, kinematics X, firing rates Z, and α.
output: learned basis U.
begin
    U ⟵ U_0
    repeat
        learn new set of linear decoding coefficients based on new U.
        F ⟵ U^T X Z^†
        learn a new set of basis vectors U that minimize the objective
        function.
        [U, ζ] ⟵ BFGS(X,Z,U,F,α)
        make sure U is orthonormal.
        U ⟵ orth(U)
    until change in the objective function value ζ < threshold
end
```

Algorithm 1: Linear Optimization

To begin the algorithm we initialize the linear subspace $\mathbf{U}$ to be equal to the first p principal components $\mathbf{U_0}$ computed on the kinematics $\mathbf{X}$. The linear optimization algorithm essentially rotates this set of orthonormal basis vectors in order to maximize eqn. 11. In each iteration the algorithm switches back and forth between computing the linear decoding coefficients $\mathbf{F}$ given the current estimate for the subspace $\mathbf{U}$ using eqn. 8, and computing a new set of basis vectors $\mathbf{U}$ given the linear decoding coefficients $\mathbf{F}$ by minimizing the objective function in eqn. 11. We minimized the objective function by using the well know quasi-Newton algorithm, BFGS [12].

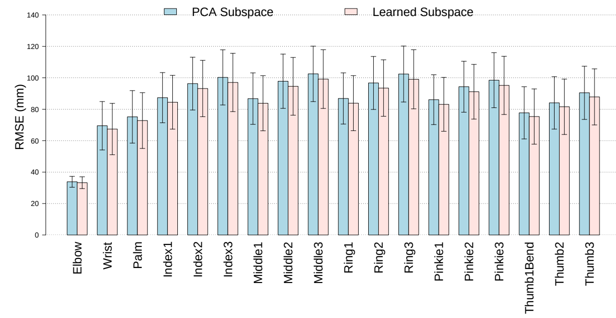


Fig. 3. RMSE of the Euclidean distance between measured and estimated positions for the 18 joints. Error bars signify one standard deviation from the mean across the 4 sessions.

VI. Experimental Setup

We tested our approach on kinematic and neural data sets collected from two rhesus macaques across multiple days. The monkeys were trained to perform reaching and grasping of 6-7 different types of objects that were held in front of them. Each session included 6-7 different trials, each consisting of 20-30 grasps of one particular object. The objects included a cylindrical pipe, triangular prism, disk, bar, small and large ball, and a pencil. For each monkey we recorded two sessions, each one on a different day. The kinematic data was captured using marker-based motion capture technology at a sampling rate of 240Hz.

Simultaneous recording of the neural population spike activity was made during the sessions via a 100-electrode array implanted in the arm/hand area of primate MI cortex. The number of cells that were classified as good units, based on high signal to noise ratio, changed for each session but were in the order of 30-150 cells. The kinematic data was then sub-sampled every 10 frames ($41.66ms$). The neural firing rates were smoothed using a Butterworth filter and a sliding $100ms$ window was used to sub-sample the neural firing rates centered every $41.66ms$.

VII. Experimental Results

Principal component analysis was conducted on the training data (first 60% of recorded frames) in each of the four recorded session separately. The kinematic data $\mathbf{X}$ consisted of the 32 exponential map/twist DoF ($d = 32$) found in the hand and arm. The first six principal components ($p = 6$), ranked in order of decreasing eigenvalues, where kept for the initial estimates of the linear subspace $\mathbf{U}_0$, for each session.

We learned a more optimal basis $\mathbf{U}$ for each of the four sessions, using the optimization framework we developed earlier. In order to optimize for the best value for the reconstruction error weighting parameter α we ran the optimization algorithm for values of $\alpha = [0, 1]$, and found the optimal value to be $\alpha \approx 0.6$.

When $\alpha = 0.6$ the first three of the six basis vectors for the optimal learned subspace $\mathbf{U}$ over one of the four sessions is shown in Fig. 1b. For the same data set, the plots of the first three optimal basis vectors show that they capture much more elbow and shoulder movement than the principal components shown in Fig. 1a. This suggests that the linear decoder is much better at decoding the shoulder and elbow rotations than the joints of the hand; possibly because there is more information in the recorded neural data relating to the DoF's in the arm than the hand. The amount of total variance in hand and arm posture explained by the learned subspace for the four sessions is shown as red dots in Fig. 2. From this we can see that the total variance explained by the learned subspace is less that the PCA subspace for all four sessions. This is interesting because we show that we can decode better in a subspace that captures much less variance in the hand and arm posture.

To have a better understanding of estimation accuracy in decoding in the two subspaces we compare the decoding accuracy in terms of reconstruction results of the joint positions in 3D Euclidean space shown in Fig. 3.

In order to compute the statistical significance between decoding in the (1) PCA subspace vs. (2) the learned subspace $\mathbf{U}$ we used the following method. Given that the error between measured and decoded joint positions in Euclidean space at time i is $e_i^{(1)}$ for case (1) and $e_i^{(2)}$ for case (2), we are interested in the error difference $d_i^{(2)-(1)}$. We found that the average portion of time that the decoding in the learned subspace did better than in the PCA subspace was 57.72 per cent, for 18 joints. The average improvement in error was $2.66\ mm$.

Given the random variable $d^{(2)-(1)}$ for each of the joints, and assuming a Gaussian distribution, we propose the null hypothesis that $d^{(2)-(1)}$ is normally distributed with mean zero, and test against the null hypothesis that the mean is not zero. For a significance level of 0.01, the results of the student t-test showed a rejection of the null hypothesis for the 18 joints, with an average confidence interval of $[-2.930 \quad -2.383]\ mm$.

VIII. Conclusions and Discussion

We have presented an optimization algorithm which takes into account neural motor control activity to learn an optimized subspace for linear decoding of kinematic postures of the hand and arm. We showed that learning a linear decoding model relating the neural firing rates and the projection of hand and arm postures into the learned subspace, as opposed to the PCA subspace, produced statistically significant improvements in posture estimates. The main contribution of this work has been to devise an algorithm that learns a set of motor "primitives" that linearly relate to the neural motor control activity.

As future work we are interested in learning jointly optimized subspaces for both the kinematic and neural data, by applying algorithms such as Canonical Correlation Analysis (CCA) and Dynamic Coupled Component Analysis [10]. The goal is to find a pair of subspaces that maximally correlate the kinematics and neural activity and provide more accurate and efficient decoding algorithms.

We are also looking into relaxing some of the assumptions we have made about our current model. Currently we are assuming that the distribution of the projection of the kinematic data onto the subspace is jointly Gaussian, which would result in independent principal components. Since the joint distribution is not exactly Gaussian this independence assumption is violated. For this reason we will investigate using Independent Component Analysis (ICA) as an initialization method for our current algorithm, instead of PCA, and adding a sparseness prior on the components in our objective function. The idea behind the sparseness prior is to find a set of components that are maximally non-Gaussian, and therefore independent, due to the central limit theorem.

Finally, we are interested in evaluating decoding accuracy for non-linear extensions to the current method by either using non-linear dimensionality reduction techniques to learn a new mapping between kinematic spaces, or by employing non-linear decoding methods such as the extended Kalman filter.

References

[1] L. R. Hochberg and M. D. S. et al, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature*, vol. 442, no. 7099, pp. 164–171, 2006.

[2] W. Wu, Y. Gao, E. Bienenstock, J. P. Donoghue, and M. J. Black, "Bayesian population coding of motor cortical activity using a kalman filter," *Neural Computation*, vol. 18, pp. 80–118, 2005.

[3] N. G. Hatsopoulos, Q. Xu, and A. Amit, "Encoding of movement fragments in the motor cortex," *Journal of Neuroscience*, vol. 27, no. 19, pp. 5105–5114, 2007.

[4] J. N. Ingram, K. P. Kording, I. S. Howard, and D. M. Wolpert, "The statistics of natural hand movements," *Experimental Brain Research*, vol. 288, pp. 223–236, 2008.

[5] C. R. Mason, J. E. Gomez, and T. J. Ebner, "Hand synergies during reach-to-grasp," *The Journal of Neurophysiology*, vol. 86, no. 6, pp. 2896–2910, 2001.

[6] E. Todorov and Z. Ghahramani, "Analysis of the synergies underlying complex hand manipulation," in *Conf. of IEEE EMBS*, 2004, pp. 4637–4640.

[7] C. E. Lang and M. H. Schieber, "Human finger independence: Limitations due to passive mechanical coupling versus active neuromuscular control," *Journal of Neurophysiology*, vol. 92, pp. 2802–2810, 2004.

[8] C. Hager-Ross and M. H. Schieber, "Quantifying the independence of human finger movements: Comparisons of digits, hands, and movement frequencies," *Journal of Neuroscience*, vol. 20, pp. 8542–8550, 2000.

[9] M. Santello, M. Flanders, and J. F. Soechting, "Patterns of hand motion during grasping and the influence of sensory guidance," *Journal of Neuroscience*, vol. 22, pp. 1426–1435, 2002.

[10] F. De la Torre and M. J. Black, "Dynamic coupled component analysis," in *CVPR*, 2001.

[11] F. S. Grassia, "Practical parameterization of rotations using the exponential map," *The Journal of Graphics Tools*, vol. 3, no. 3, pp. 29–48, 1998.

[12] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, August 1999.