Asynchronous Decoding of Dexterous Finger Movements Using M1 Neurons

Vikram Aggarwal, *Student Member, IEEE*, Soumyadipta Acharya, *Student Member, IEEE*, Francesco Tenore, *Member, IEEE*, Hyun-Chool Shin, Ralph Etienne-Cummings, *Member, IEEE*, Marc H. Schieber, and Nitish V. Thakor, *Fellow, IEEE*

***Abstract*—Previous efforts in brain–machine interfaces (BMI) have looked at decoding movement intent or hand and arm trajectory, but current cortical control strategies have not focused on the decoding of 3 actions such as finger movements. The present work demonstrates the *asynchronous* decoding (i.e., where cues indicating the onset of movement are not known) of individual and combined finger movements. Single-unit activities were recorded sequentially from a population of neurons in the M1 hand area of trained rhesus monkeys during flexion and extension movements of each finger and the wrist. Nonlinear filters were designed to detect the onset of movement and decode the movement type from randomly selected neuronal ensembles (assembled from individually recorded single-unit activities). Average asynchronous decoding accuracies as high as 99.8%, 96.2%, and 90.5%, were achieved for individuated finger and wrist movements with three monkeys. Average decoding accuracy was still 92.5% when combined movements of two fingers were included. These results demonstrate that it is possible to asynchronously decode dexterous finger movements from a neuronal ensemble with high accuracy. This work takes an important step towards the development of a BMI for direct neural control of a state-of-the-art, multifingered hand prosthesis.**

***Index Terms*—Brain–machine interface (BMI), dexterous control, neural decoding, neural interface, neuroprosthesis.**

I. Introduction

A BRAIN–MACHINE interface (BMI) uses electrophysiological measures of brain function to enable communication between the brain and external devices, such as computers or mechanical actuators. Studies using intracortical BMIs have demonstrated the decoding of neuronal ensemble activity in the dorsal premotor cortex (PMd) [1], [2] primary motor cortex (M1) [3]–[6], and posterior parietal cortex (PPC) [7], [8] for the purposes of deriving a variety of cortical control signals.

Previous work using signals from the rat motor cortex has shown 1-D control of a robotic arm using a population of 32 M1 neurons to predict hand trajectory [3], [4]. It has also been shown that simultaneous activity from the PMd, M1, and PPC areas of nonhuman primates could accurately predict continuous 3-D arm movement trajectories [9]. Additional research in nonhuman primates has shown that activity from even relatively few neurons in M1 can reliably decode movement of a cursor on a computer screen [5], [6]. These results have led to breakthroughs in the translation of direct cortical control strategies from animal trials to humans. Recently, a report presented a tetraplegic human who was able to control a computer cursor to open e-mail, operate a television, and open and close a prosthetic hand using neuronal ensemble activity from a microelectrode array [5].

In general, these experiments have demonstrated the robust coding capacity of neural populations and have opened up the possibility of a BMI for direct neural control of a prosthetic limb and the restoration of motor control for amputees, paralyzed individuals, and those with degenerative muscular diseases. These efforts have focused largely on controlling a computer cursor [5], [6], predicting movement intent ([1], [8], and decoding hand and arm trajectory ([2]–[4], [9], [10], however, and current neural control strategies have not focused on *dexterous* neuroprosthetic control of actions such as individuated and combined finger movements.

Prior studies of motor cortical activities in primates by Schieber *et al.* [11]–[14] and Georgopoulos *et al.* [15] have shown that there are indeed neurons in the primary motor cortex that code for specific finger movements, and which discharge during movements of several fingers. Their results also suggest that neuronal populations active with movements of different fingers overlap extensively in their spatial locations in the motor cortex [13], [14]. Although there are neurons that fire maximally for a specific movement type, these neurons are not localized to a specific anatomic region, but rather are distributed throughout the hand region [14]. More recent MRI studies with humans also show that while there are regions of the primary motor cortex active with different finger movements, there is also an overlap of finger representation in the motor homunculus [16], [17]. The lack of strict somatotopic organization of the M1 hand area suggests that a given subpopulation of neurons in this region should contain sufficient information to encode for individuated and combined movements of the fingers and wrist.

Manuscript received August 23, 2007; revised November 30, 2007; accepted November 30, 2007. This work was supported in part by the Defense Advanced Research Project Agency (DARPA) Revolutionizing Prosthetics 2009 program and in part by from the National Institute of Neurological Disease and Stroke (NINDS) R01/R37 NS27686.

V. Aggarwal, S. Acharya, and N. V. Thakor are with the Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD 21205 USA.

F. Tenore and R. Etienne-Cummings are with the Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218 USA.

H. C. Shin is with School of Electronic Engineering, College of Information Technology, Soongsil University, 156-743 Seoul, Republic of Korea.

M. H. Schieber is with the Departments of Neurology, Neurobiology and Anatomy, Brain and Cognitive Sciences, and Physical Medicine and Rehabilitation, University of Rochester Medical Center, Rochester, NY 14642 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNSRE.2007.916289

Original work using population vectors [18], which combine the weighted contributions from each neuron along a preferred direction, found that 75% of motor cortical neurons related to finger movements were tuned to specific directions in a 3-D instructed hand movement space [15]. Although population vectors were fairly good predictors of the direction of instructed finger movements, this approach only yielded 60%–70% decoding accuracy. Using both linear and nonlinear classification schemes, such as logistic regression and softmax, Pouget *et al.* have demonstrated decoding of individuated and combined finger movements with higher accuracy [19].

This decoding, however, was accomplished in a synchronous fashion using *a priori* knowledge of the movement events, i.e., decoding based on information from only the time of movement [15], [19]. BMI control of a neuroprosthesis, on the other hand, requires *asynchronous* decoding of neuronal data in real-time, i.e., where cues indicating the onset of movement are not known. The goal of this study was to demonstrate, for the first time, the feasibility of a BMI for dexterous control of individual fingers and the wrist of a multifingered prosthetic hand by detecting movement onset as well as decoding movement type. The final decoded output then was used to actuate a multifingered robotic hand in real-time for a preplanned task.

II. Methods

A. Experimental Setup

Three male rhesus (*Macaca mulatta*) monkeys—C, G, K—were trained to perform visually-cued movements of individual fingers and the wrist. Each monkey placed its right hand in a pistol-grip manipulandum, which separated each finger into a different slot. By flexing or extending each digit a few millimeters, the monkey closed either a ventral or dorsal micro-switches that were at the end of each slot. The manipulandum was also mounted such that it allowed for flexion and extension of the wrist. A detailed description of the experimental protocol can be found in [12].

All the monkeys performed 12 distinct individuated movements: flexion and extension of each of the fingers and of the wrist of the right hand. Each instructed movement is abbreviated with the first letter of the instructed direction ($\mathrm{f} = \mathrm{flexion}$ and $\mathrm{e} = \mathrm{extension}$), and the number of the instructed digit ($1 = \mathrm{thumb} \ldots 5 = \mathrm{little\ finger}, w = \mathrm{wrist}$; e.g., “e4” indicates extension of ring finger). Monkey K performed an additional six combined finger movements involving two digits: $\mathrm{f1}+2, \mathrm{f2}+3, \mathrm{f4}+5, \mathrm{e1}+2, \mathrm{e2}+3, \mathrm{e4}+5$.

The monkey was trained to initiate successive trials by placing all digits and the wrist in their neutral positions, so that no switches were closed and the middle yellow LED in each row was lit. After a pseudo-random ready time of 500–750 ms, one red LED in a single row was illuminated instructing the monkey whether to flex or extend that digit. If the monkey closed the instructed switch within 700 ms response time of the red LED cue, and held it closed for a hold time of 500 ms, then the trial was a success and the monkey received a water reward. If the trial was a failure—either by failing to close the instructed switch within 700 ms or by closing another switch—no reward was given and the same instruction was presented again. If the trial was a success, the next instructed finger movements was presented in a pseudorandom order. Therefore, consecutive rewards for successful trials of a given movement did not occur but instead were separated by trials of other instructed movements.

The monkey was prepared for single-unit recording by surgically implanting both a head-holding device and a rectangular Lucite recording chamber that permitted access to the area encompassing M1 contralateral to the trained hand. Single-unit activities were sequentially recorded from 312 task-related neurons in M1 for monkey C, 125 neurons for monkey G, and 115 neurons for monkey K during daily 2–3-h recording sessions. These recordings were obtained using single self-made, glass-coated, Pt-Ir, microelectrodes. The duration of each trial was approximately 2 s, and for data analysis all trials were aligned such that switch closure occurred at 1 s. For monkey C and monkey G, data from each neuron for up to seven trials per movement type was used, while for monkey K data from each neuron for up to 15 trials per movement type was used.

B. Analyzing the Input Space

Different neurons showed different types of tuning to the set of movements. For example, certain neurons may be broadly-tuned and responsive to multiple movement types, whereas other neurons may be highly-tuned to specific movement types. Four examples of neurons from monkey K are illustrated in Fig. 1. The spike histograms show the number of spikes discharged in the 100 ms preceding switch closure for each movement, averaged across all trials.

Neurons K11404 and K13905 show a broad response to most movement types, while neurons K13409 and K13506 are responsive primarily to extension of the index finger. Since the input space (i.e., multidimensional feature space comprising the neuronal firing rates) will not include dedicated neurons responsive to each movement type, an appropriate mathematical model describing the neuronal firing patterns is needed to unmask the input–output relationship of this complex multidimensional space.

Fig. 2(A) further demonstrates the complexity of the input space by looking at the temporal evolution of spiking activity of a population of neurons in monkey K, averaged across all trials. As seen in Fig. 2(B), there was an increase in neuronal activity in the period directly preceding the switch closure at 1 s (shown for 12 neurons and four movement types). Since the neuronal activity is dynamic, any model describing the input space must account for changes in the neuronal activity during periods of rest and during periods of movement. Therefore, to make the problem tractable, the input space was divided into disjoint subspaces corresponding to either rest or finger movement. The first subspace resolved the onset of movement from baseline activity, while the second subspace decoded movement type from the period of activity directly preceding switch closure.

C. Choosing a Mathematical Model

Given the complex nature of neuronal firing patterns, deciphering the neural coding associated with dexterous tasks such as individual finger movements is not trivial. Many neural decoding algorithms have been developed, the simplest of which is a linear estimator that has demonstrated effective neural control of a cursor [6] or arm trajectory [2], [10], [20]. More recently, probabilistic approaches such as Kalman filter and maximum

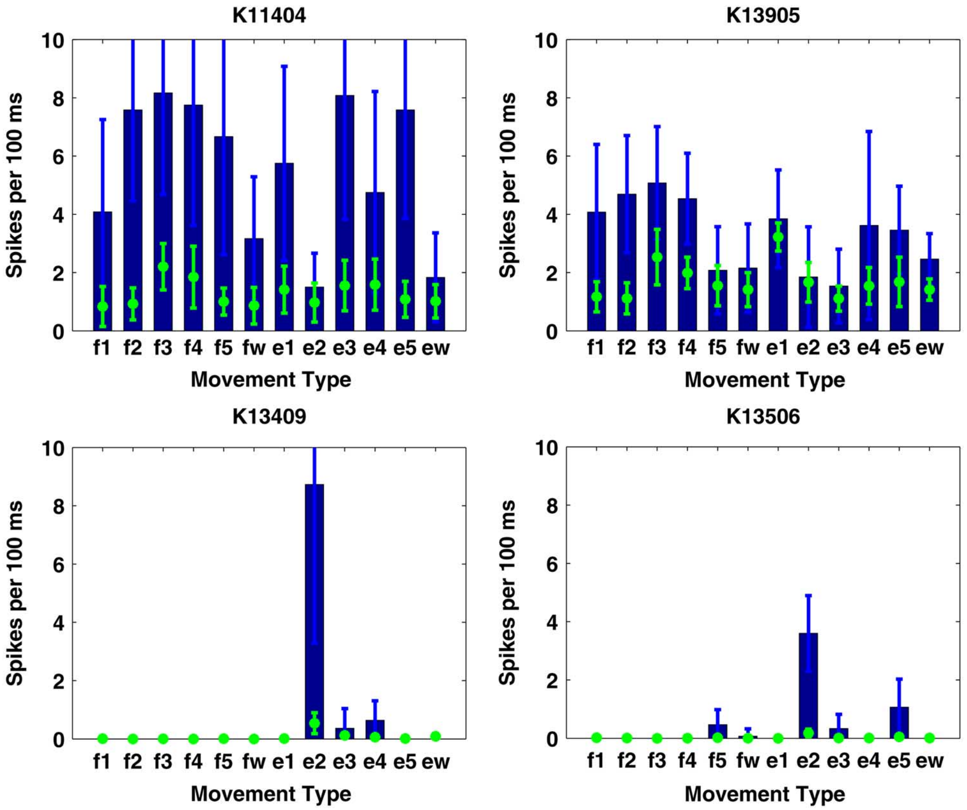

Fig. 1. Histograms of spike counts from sample neurons in monkey K show that the neuronal population contains both broadly-tuned neurons active during multiple finger movements (e.g., K11404, K13905), and highly-tuned neurons active during only specific finger movements (e.g., K13409, K13506). Histograms represent spiking activity during the 100 ms movement period directly preceding switch closure, while the scatter points represent background activity during nonmovement periods. Spike rates are averaged across all trials.

likelihood (ML) have also been exploited for neural control of a 2-D cursor [21], [22] and decoding reaching movements [1], [23], [24], respectively. These models, however, assume that each neuron can be treated as an independent unit or are uncorrelated, but given the spatial distribution of activity across various finger movement types it is unclear whether this assumption holds true for finger movements.

Therefore, nonlinear decoding filters were designed using multilayer, feedforward artificial neural networks (ANNs), which have been widely used in nonlinear regression, function approximation, and classification [25], and have become a popular approach for neural decoding of movements [2]–[4]. In order to decode individual and combined finger movements in real-time, a hierarchical classification scheme was developed. A Gating Classifier was designed to distinguish baseline activity from the onset of movement, while a Movement Classifier was designed to distinguish amongst the individual or combined movement types. Both classifiers were designed as committee neural networks, which are commonly used for classification of biological signals and known to improve decoding accuracy [26].

D. Detecting Onset of Movement: Gating Classifier

The Gating Classifier was designed as a committee of N neural networks (where N was odd), and trained to distinguish baseline activity from movement. Each ANN in the committee was trained using the neuronal firing rate over a sliding temporal window of duration T_W that was shifting every T_S (chosen to be 100 ms and 20 ms, respectively, ANNs were designed with n input neurons (where n is the number of sampled neurons), and a single output neuron. The output was normalized to between 0 and 1.

In order to train the gating networks appropriately, however, it was important to first look at the temporal evolution of spiking activity. Fig. 3 shows example time courses of spike activity for a single neuron from monkey K. The plot overlays single trial data for each of the twelve individual finger movement types. A reasonable approach would have been to simply assign the point of switch closure as having an output of 1 and all other time points to have an output of 0 [Fig. 3(A)]. However, due to the fact that 1) this would result in limited training points and 2) the neuronal activity gradually evolved and reached a peak even prior to switch closure, the networks were trained instead with a trapezoidal membership function [Fig. 3(B)] and assigned a fuzzy output label, $P_I(t_k)$ at discrete times t_k. The membership function parameters, $t_r, t_f, t_1, t_{2,}$ were determined empirically and shown in Fig. 3(B) (inset)

$$P_I(t_k) = \begin{cases} \frac{T_s}{t_1 - t_r} \cdot t_k, & \text{if } t_r \le t_k < t_1 \\ 1, & \text{if } t_1 \le t_k < t_2 \\ \frac{-T_s}{t_f - t_2}(t_k - t_f), & \text{if } t_2 \le t_k < t_f \\ 0, & \text{else.} \end{cases} \quad (1)$$

Since the output values ranged between 0 and 1, they could be interpreted as a probability for movement activity. The output of

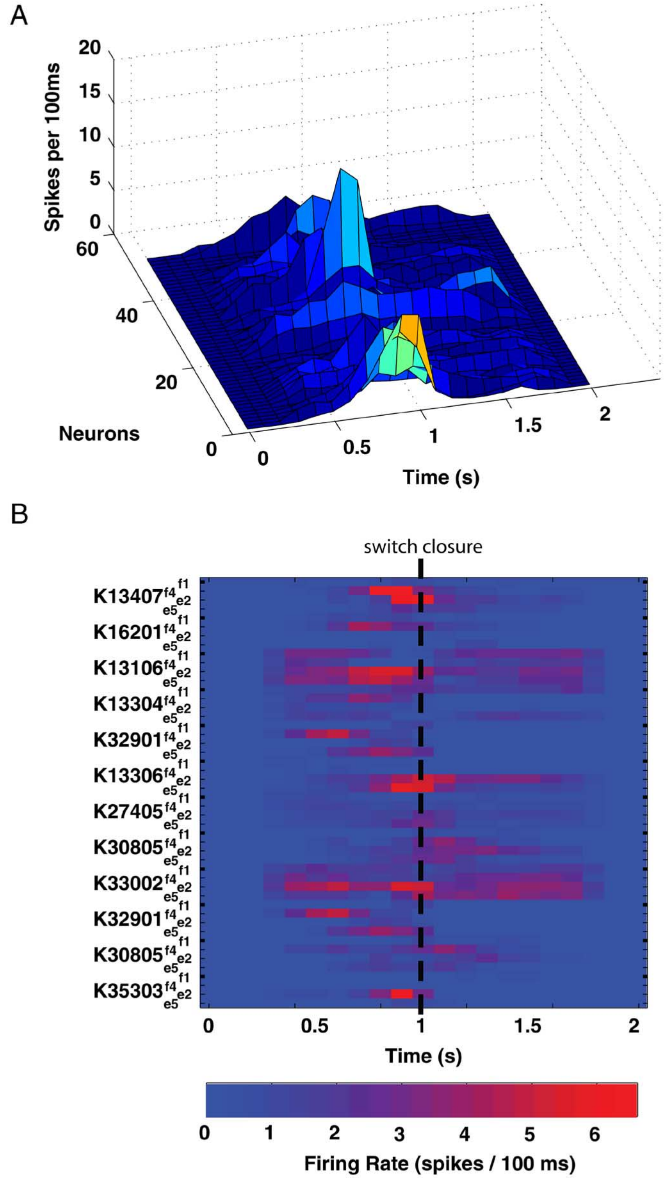

Fig. 2. A: Time course of neuronal activity from monkey K shows a gradual evolution of spiking activity around the time of switch closure at 1 s. B: Although a clear increase in activity is evident, the effect varies across neurons and movement types (shown for 12 sample neurons from monkey K and 4 movement types), which adds to the complexity of the problem at hand. Spike rates are averaged across all trials.

each individual gating network, $\hat{P}_I(t_k)$, was then thresholded at a value γ to produce a discrete binary variable

$$g(t_k) = \begin{cases} 1, & \text{if } \hat{P}_I(t_k) > \gamma \\ 0, & \text{else} \end{cases} \tag{2}$$

where an output of 0 corresponded to rest and 1 corresponded to movement. A majority voting rule was then used to determine the final committee output, $G(t_k)$

$$G(t_k) = \sum_{n=1}^{N} (g_n(t_k)) > \frac{N}{2}. \tag{3}$$

In order to eliminate spurious gating classifications, the committee output was tracked continuously. The Gating Classifier, $G_{\text{track}}(t_k)$, would only fire if the committee output indicated movement activity at least β times over the previous τ time points (where $\beta < \tau$),

$$G_{\text{track}}(t_k) = \begin{cases} 1, & \text{if } \sum_{t=t_k-\tau}^{t_k} (G(t_k)) > \beta \\ 0, & \text{else.} \end{cases} \tag{4}$$

Furthermore, once the Gating Classifier fired, a refractory period, ρ, prohibited the classifier from firing again immediately thereafter. γ, β, τ, and ρ were constants optimized to maximize classifier accuracy and empirically determined to be $\gamma = 0.7, \beta = 7, \tau = 10$, and $\rho = 125$ ms.

E. Decoding Movement Type: Movement Classifier

The Movement Classifier also was designed as a committee of N neural networks (where N was odd), but was trained to distinguish only amongst each movement type. Each ANN in the committee was trained using the neuronal firing rate during only the 100 ms window preceding switch closure (T_{switch}). ANNs were designed with n input neurons (where n is the number of sampled neurons, and identical to those used for the Gating Classifier), and m output neurons (where m is the number of movement types). The output was normalized to between 0 and 1.

The networks were trained with a binary membership function [Fig. 3(C)] and assigned an output label for each movement type, $P_{M,i}(t_k)$ at discrete times t_k (where $i = 1, \ldots, 12$ corresponding to the 12 individuated movement types and $i = 13, \ldots, 18$ corresponding to the six combined movement types). For the ith movement type, the ith output neuron was assigned an output of 1 during this window and all other neurons were assigned an output of 0

$$\begin{aligned} P_{M,1}(t_k) &= 0 \text{ if } (T_{\text{switch}} - 0.1) \le t_k < T_{\text{switch}} \\ &\vdots \\ P_{M,i}(t_k) &= 1 \text{ if } (T_{\text{switch}} - 0.1) \le t_k < T_{\text{switch}} \\ &\vdots \\ P_{M,m}(t_k) &= 0 \text{ if } (T_{\text{switch}} - 0.1) \le t_k < T_{\text{switch}}. \end{aligned} \tag{5}$$

Once again, since the output values ranged between 0 and 1, they could be interpreted as probabilities for a specific movement type. The output of each of the individual movement networks, $\hat{P}_{M,i}(t_k)$, was selected based on the output neuron that had the greatest output activity (or probability) to produce

$$s(t_k) = \arg\max_{i} \ \hat{P}_{M,i}(t_k)\}. \tag{6}$$

A majority voting rule was then used to determine the final committee output of the movement classifier, $S(t_k)$,

$$S(t_k) = \text{mode}\{s_n(t_k)\}. \tag{7}$$

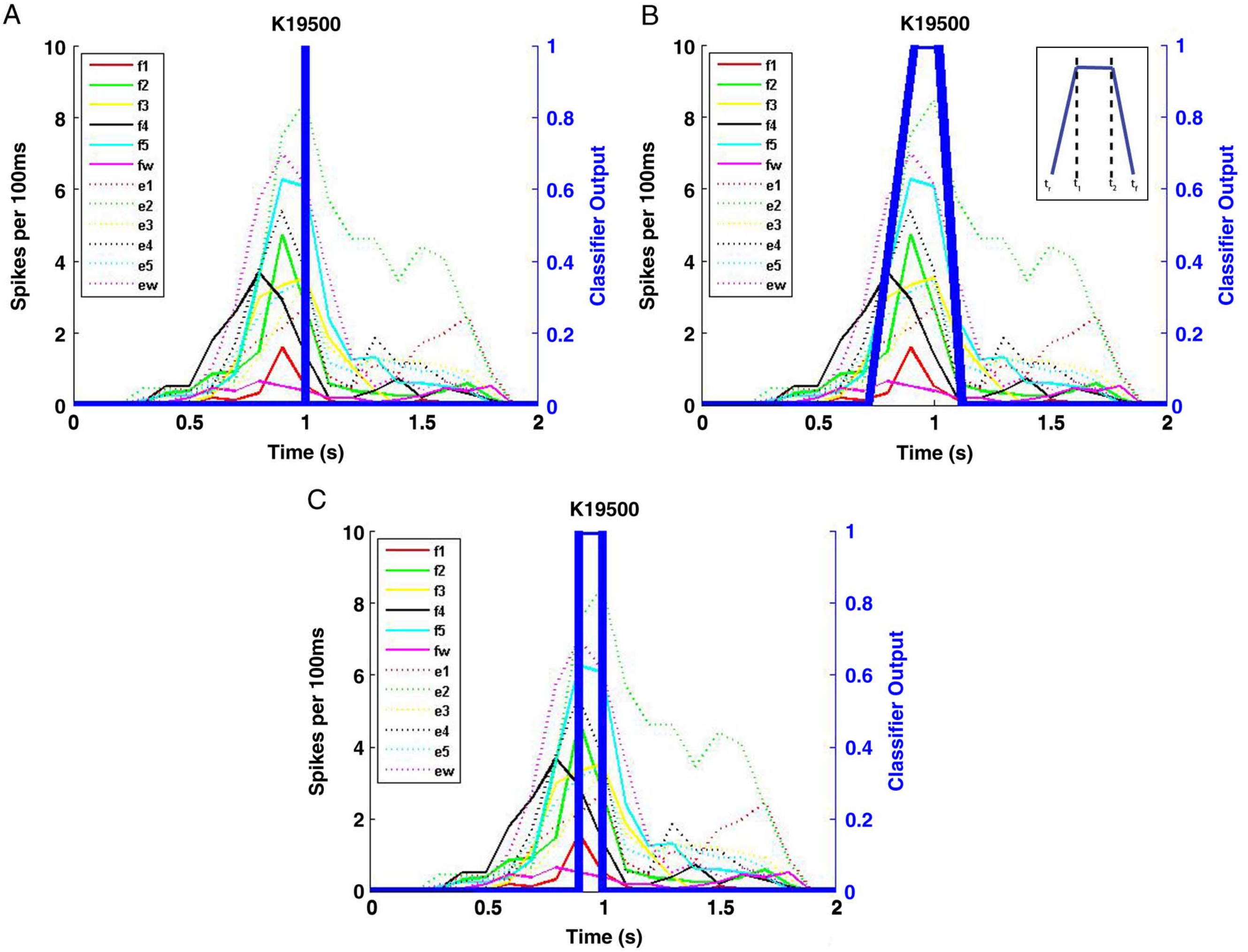

Fig. 3. Time course of a sample neuron in monkey K (K19500) during movement of individual fingers shows that the neuronal firing rate gradually increased and reached a peak at (or slightly before) the time of switch closure at 1 s. A: Rather than training the Gating Classifier at only the time of switch closure. B: "Soft" trapezoidal membership function was used to train the classifier over a broad time period. Sensitivity of the Gating Classifier was improved by applying fuzzy decision boundaries and thresholding the output of the classifier. C: To train the Movement Classifier, however, only the neuronal firing rate during the 100 ms period of activity directly preceding switch closure was used. Therefore, the Movement Classifier only decoded neuronal activity associated with the actual movement of a specific finger, and not just the onset of movement.

In the absence of a simple majority, the output of the individual movement network that demonstrated the highest accuracy (on the validation set) was selected.

F. Combining the Classifiers

A total of five gating networks and five movement networks were trained for any given number of neurons. The networks differed by virtue of different training sets and a different number of hidden layer neurons. The networks were ranked based on decoding accuracy of an independent validation set, and the top three networks were selected to form the gating and movement classifier committees, respectively.

After applying a majority voting rule to get the final Gating Classifier output, $G_{\mathrm{track}}(t_k)$, and Movement Classifier output, $S(t_k)$—the final decoded output, $F(t_k)$, was obtained from the product of the two classifiers. Fig. 4 shows a block diagram summarizing this hierarchical classification scheme

$$F(t_k) = G_{\mathrm{track}}(t_k) \times S(t_k). \tag{8}$$

Individual 2 s trials were concatenated to form neuronal spike trains with activity from all the neurons selected. A classification decision was made every 20 ms using the previous 100 ms of spike activity. A decoded output of 0 yielded no action; 1-6 led to flexion of the respective digits and the wrist $(\mathrm{f1}, \ldots, \mathrm{f_w})$; and 7–12 led to the corresponding extension movements $(\mathrm{e1}, \ldots, \mathrm{ew}$ For combined finger movements, an output of 13–15 led to flexion of multiple digits $(\mathrm{f1+f2},\ \mathrm{f2+f3},\ \mathrm{f4+5})$ and an output of 16–18 led to extension of multiple digits $(\mathrm{e1+e2},\ \mathrm{e2+e3},\ \mathrm{e4+e5})$.

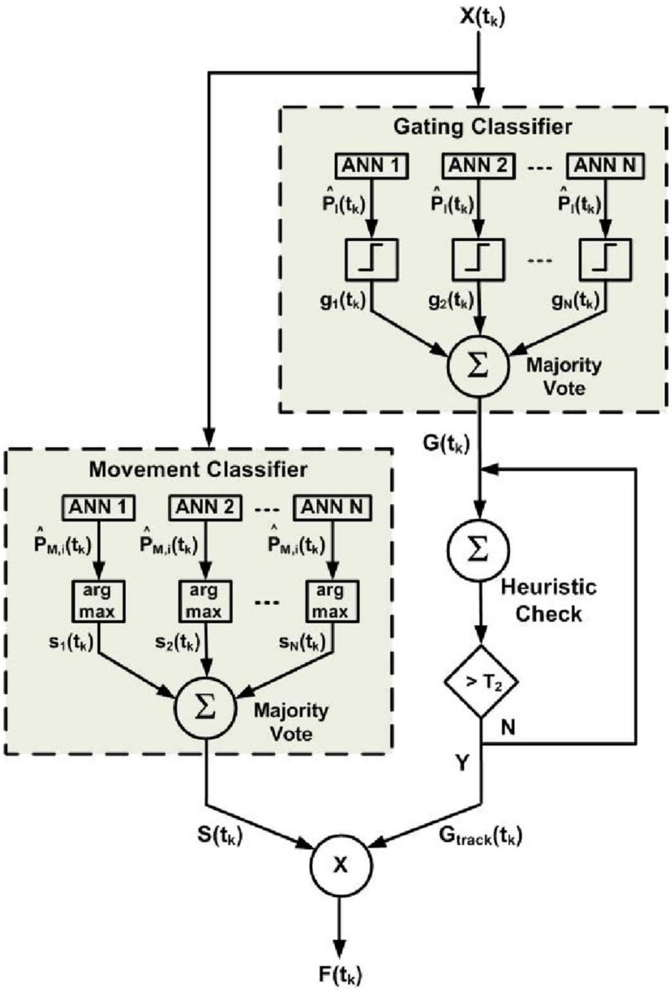

Fig. 4. Hierarchical classification scheme was developed to decode individuated and combined finger movements in real-time. Gating Classifier distinguished baseline activity from movement intent, while a Movement Classifier distinguished amongst the individual or combined movement types. represents the output of each trained gating network, $g(t_k)$ is the threshold output of each gating network, $G(t_k)$ is the committee output (after majority voting), and $G_{\text{track}}(t_k)$ is the final output of the Gating Classifier after our heuristic wrapper to eliminate spurious classifications. Similarly, $\hat{P}_{M,i}(t_k)$ represents the output of the trained movement network for each movement, $s(t_k)$ is the consolidated output of each movement network (determined by the most active output neuron), and $S(t_k)$ is the final committee output (after majority voting). Each classifier was designed as a committee of N neural networks ($N = 3$), and the final decoded output, $F(t_k)$, was the product of the two committee networks.

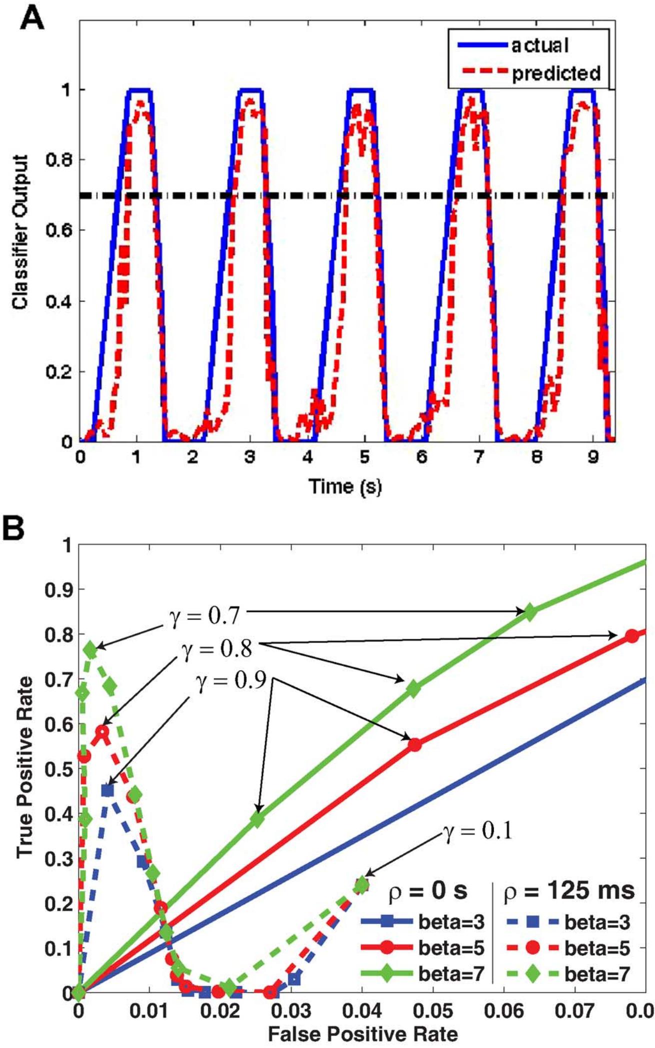

Fig. 5. A: Raw output from a single gating network (using 20 neurons) shows that the predicted output (dotted line) tracked the actual output (solid line) very closely. Evidently, the Gating Classifier performed well in decoding movement occurrence. B: ROC curves for the Gating Classifier (using 20 neurons) show the effect of the threshold parameter (γ), smoothing parameter (β), and refractory period (ρ) on reducing spurious gating classifications. When a refractory period (i.e., period immediately following decoded movement when Gating Classifier is prevented from firing again) is used, the ROC curve reaches a peak at an optimal threshold value for a given b (i.e., minimum number of positive gating decisions required for Gating Classifier to fire). Optimal selection of the smoothing ($\beta = 7$) and threshold ($\gamma = 0.7$) parameters result in a high true positive rate and low false positive rate.

G. ANN Training

All ANNs were designed with a single hidden layer containing 0.5–2.5 times the number of input neurons, tan sigmoidal transfer functions for the hidden layer (to ensure continuity and derivability throughout the output space), and log sigmoidal transfer functions for the output layer (to contain the output values within a fixed range $[0 \quad 1]$. Dimensionality of the input space was reduced using principle component analysis (PCA). After rank-ordering the components by variance, only those principle components that cumulatively contributed to $> 95\%$ of the total variance were retained. The networks were trained using the scaled conjugate gradient (SCG) algorithm [27], and early validation stop to prevent overfitting. Training, validation, and testing data were selected from mutually exclusive trials—with $1/2$ used for training, $1/4$ used for validation, and $1/4$ used for testing. Bootstrapped training, validation, and testing sets were created by randomly permuting trials across each neuron. The neural networks were trained offline using MATLAB 7.4 (Mathworks Inc., Natick, MA) Separate decoding filters were designed for each neuronal subpopulation selected.

H. Real-Time Actuation of Multifingered Prosthetic Hand

The final decoded output was used to drive a multifingered robotic hand, which permitted independent flexion/extension of each finger. Each joint was pulse-width modulated using independent servo motors to determine the final joint angle. The real-time decoded output stream was transmitted wirelessly from

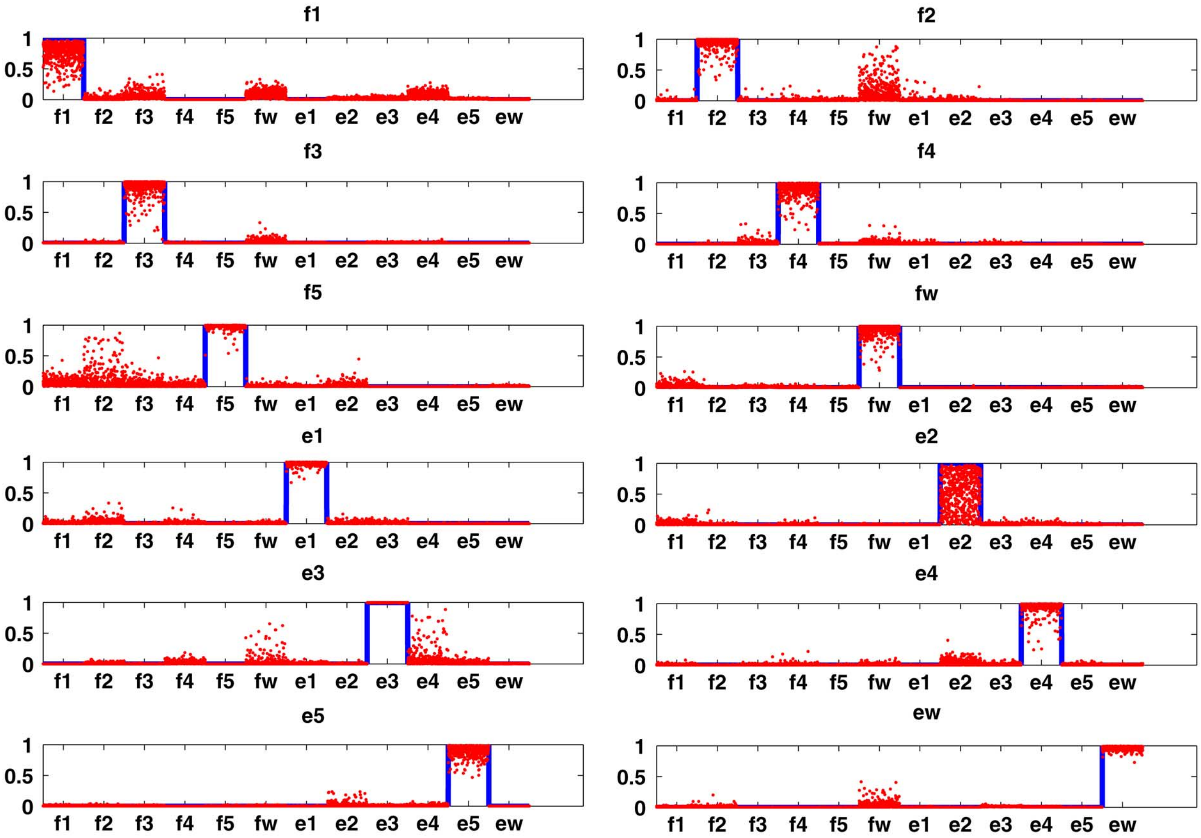

Fig. 6. Raw output from a single movement network shows that the predicted output neuron activity (scatter points) specific to the movement type to be decoded (solid line) Evidently, the movement classifier performed well in decoding the movement type. Each subplot shows the classifier output corresponding to 100 testing sets for each movement type.

a standalone PC running the signal processing algorithms to a custom 16-bit PIC microcontroller (Microchip Technology, Inc., Chandler, AZ) that interfaced with the individual servo motors. Asynchronously decoded commands from the microcontroller were then used to actuate individual fingers of the robotic hand. By synthesizing this sequential output, it was possible for the robotic fingers to execute a coordinated sequence of movements and play musical tunes on a piano keyboard in real-time.

III. Results

Asynchronous decoding of complex movements requires simultaneous recordings from an ensemble of neurons. Therefore, simultaneous multiunit activity was simulated from the single-unit activities, event-locked to time of switch closure. In this manner, individually recorded neural data were assembled together to form an ensemble. Subpopulations of neurons were randomly selected from the total population of recorded task-related neurons (312 neurons for monkey C; 125 neurons for monkey G; 115 neurons for monkey K). Results were calculated for 100 testing sets per movement type, and the decoding accuracy was averaged across six random subsets of a given number of neurons.

A. Classifier Output

Fig. 5(A) shows a samplew7 raw output of a single gating network. As can be seen, the predicted output closely tracked the actual output, indicating that the Gating Classifier detected the occurrence of a movement well. From (2), a binary output was created by thresholding the output of the gating network at γ (dotted line).

Fig. 5(B) shows how the true positive rate and false positive rate of the Gating Classifier vary as a function of the threshold parameter (γ), smoothing parameter (β), and refractory period (ρ). When there is no refractory period, the false positive rate and true positive rate asymptotically approach unity as the threshold decreases. However, with a refractory period, the ROC curve reaches a peak at an optimal threshold value for a given β. As can be seen, both optimal smoothing $(\beta = 7)$ and thresholding $(\gamma = 0.7)$ of the classifier output results in a dramatic decrease in the false positive rate without compromising the true positive rate.

Fig. 6 shows the sample output of a single movement network (in this case trained for only 12 individual movements). Each subplot shows the output for the 12 output neurons corresponding to 100 testing sets for each movement type. As can be seen, for the most part each of the output neurons was only active for the specific movement type that it was trained to recognize. From (6), the final output was selected based on the output neuron that had the greatest output activity. Each subplot shows the classifier output corresponding to 100 testing sets for each movement type.

To illustrate the benefits of a committee network, Figs. 7(A)–(C) shows the final decoded output for three different classifiers where only a single gating network and

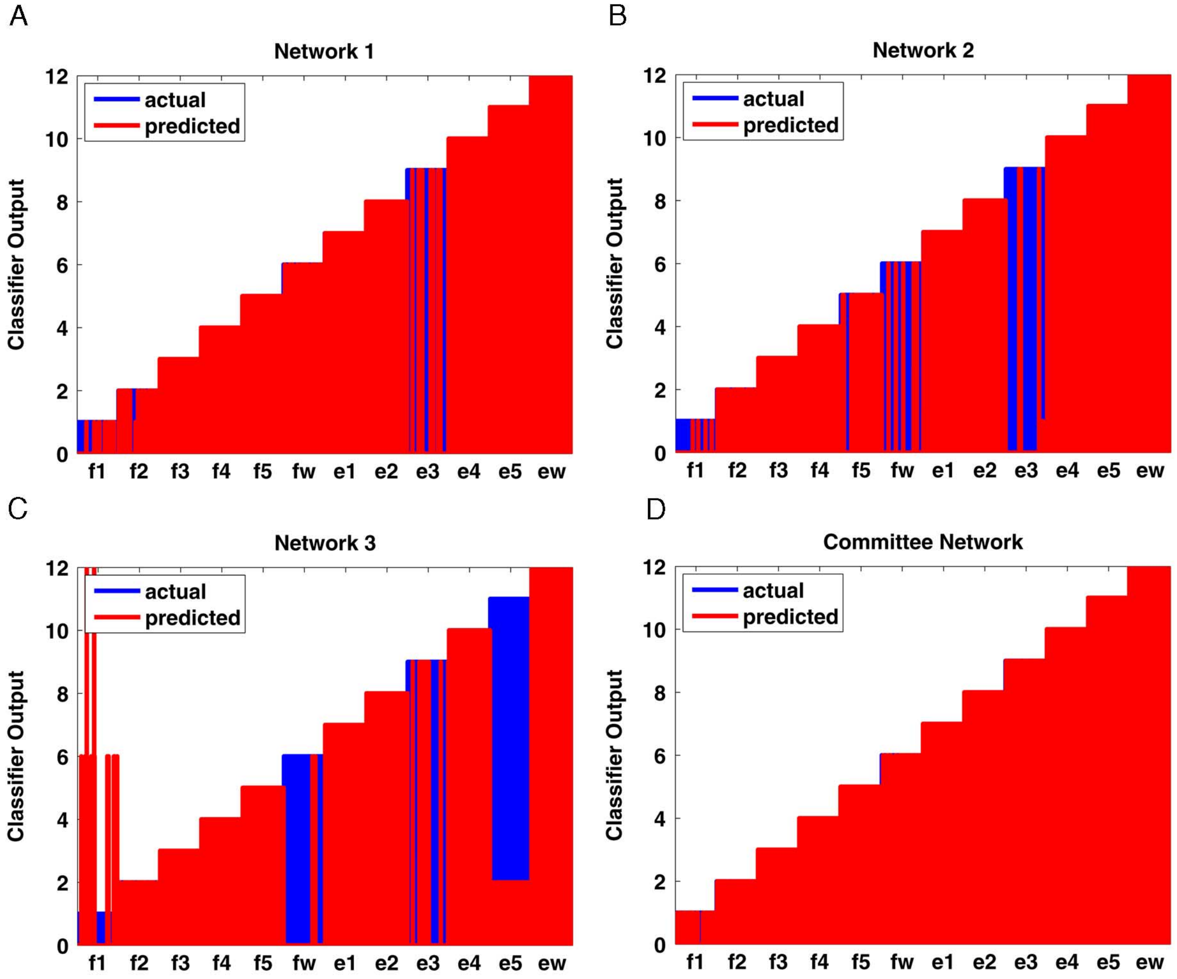

Fig. 7. A-C: shows the final decoded output for three different classifiers where only a single gating and movement network were used (shown for monkey K using 20 neurons for decoding). Each classifier fared poorly individually, but when a committee network is employed (D) the overall final decoded output showed a marked improvement. Each subplot shows the classifier output corresponding to 100 testing sets for each movement type.

movement network were used. Each subplot shows the classifier output corresponding to 100 testing sets for each movement type. Fig. 7(D) shows the final decoded output when a committee network was employed instead. As can be seen, the committee improved decoding accuracy markedly.

B. Asynchronous Decoding Accuracy

Fig. 8 shows the average asynchronous decoding accuracy across the 12 individuated finger movements for monkey C, monkey G, and monkey K, as well as across the 18 individuated and combined finger movements for monkey K. Using 40 neurons, the average decoding accuracy for individuated finger and wrist movements was as high as 99.8% for monkey K, 96.2% for monkey C, and 90.5% for monkey G. Using only 25 neurons, accuracy was still 95.4% for monkey K, 85.0% for monkey C, and 72.9% for monkey G. When combined finger movements of two fingers were included, the average decoding accuracy was still as high as 92.5% with as few as 40 neurons for monkey K. These results are summarized in Table I.

Asynchronous decoding accuracy also was examined separately for each movement type to determine whether accuracy varied for different movements. Fig. 9 shows the decoding accuracy for the 12 individuated finger movements for monkey C, monkey G, and monkey K, as well as the 18 individuated and combined finger movements for monkey K. The results show that 10 out of 12 movement types are decoded with greater than 95% accuracy using 50 neurons for monkey C, with flexion and extension of the wrist being two of the movements that were decoded poorly. Twelve out of 12 movement types, however, are decoded with greater than 99% accuracy using 50 neurons for monkey K. When the six additional combined movements are included, 13 out of 18 movement types are decoded with greater than 99% accuracy using 50 neurons for monkey K.

IV. DISCUSSION

The results demonstrate that activity from an ensemble of randomly-selected, task-related neurons in the M1 hand area contains sufficient information to allow for the asynchronous decoding of individuated movements of the fingers and the wrist. Average asynchronous decoding accuracies for individuated movements approached 100% for two of the three monkeys with 40 neurons, and were still greater than 90% for the remaining monkey or when including combined finger

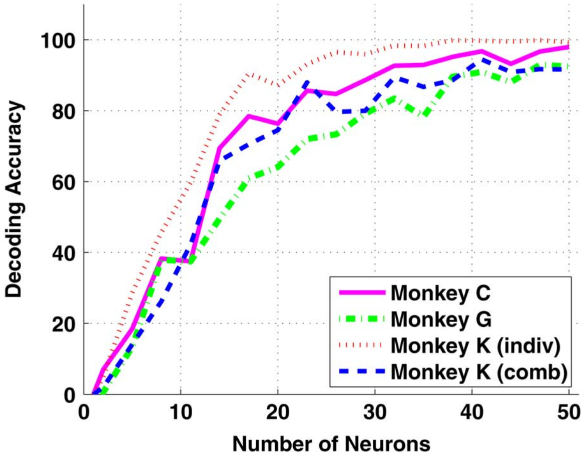

Fig. 8. Asynchronous decoding results for individuated and combined finger movements. For individuated movements, decoding accuracy was as high as 99.8% for monkey K using 40 neurons, and 95.4% using only 25 neurons. Although lower, decoding accuracy was still 96.2% for monkey C and 90.5% for monkey G using 40 neurons. When combined movements were included, average decoding accuracy was 92.5% for all 18 movement types using 40 neurons for monkey K.

TABLE I
SUMMARY OF ASYNCHRONOUS DECODING RESULTS

Monkey	25 Neurons	35 Neurons	40 Neurons
C	85.0% ± 3.0%	92.9% ± 1.3%	96.2% ± 1.8%
G	72.9% ± 2.1%	78.3% ± 3.3%	90.5% ± 2.1%
K (indiv)	95.4% ± 1.0%	98.3% ± 1.5%	99.8% ± 0.1%
K (comb)	82.4% ± 2.1%	86.7% ± 1.6%	92.5% ± 1.1%

* *results reported as accuracy* ±*SE*

movements. In particular, monkeys C and K achieved average decoding accuracies of greater than 90% with as few as 28 neurons for monkey C and 17 neurons for monkey K. Monkey G was a relatively poor performer of the individuated movement task (unpublished observations from Schieber), which may explain why overall decoding accuracies were lower for monkey G than the other monkeys.

The consistently high overall decoding accuracies across multiple monkeys, each with a different distribution of neural recordings in the M1 hand area, illustrate the robustness of the decoding algorithm to successfully decode this complex multidimensional input space irrespective of the specific neuronal population. However, it is apparent that there are biases in decoding accuracies when looking at specific movement types. Although flexion of the thumb and flexion/extension of the wrist are typically the easiest movements for monkeys to perform, these movements (f1, fw, ew) had the poorest decoding performance (see Fig. 9). Previous reports show less neural activity during f1, fw, and ew, than during other movements [14]. Movements that are simpler for the monkey to perform thus may entail less M1 neural activity, making such movements paradoxically more difficult to decode.

A. *Novelty of Decoding Algorithm*

Unlike previous studies [15], [19] that have focused on decoding finger movements, our results are the first to demonstrate asynchronous decoding where both the onset of movement and movement type are decoded. This novel hierarchical classification scheme is well-suited for BMI applications where cues indicating the onset of movement are not known. Although the training of neural networks is computationally intensive, the decoding filters are trained offline and the algorithm can still be applied to real-time applications with little latency.

Furthermore, our results show decoding accuracies on par or better than previous synchronous decoding approaches [15], [19]. In addition to ANNs capable of resolving this highly complex input space, a committee neural network approach was used to increase the decoding accuracy and reduce the associated measure of uncertainty. It has been shown that a committee of classifiers, each of which performs better than random chance, can result in remarkably higher classification accuracies than the individual classifiers themselves [26]. Even though one network may have decoded a movement incorrectly, on any given trial the remaining networks in the committee may have decoded the movement correctly. Therefore, based on a majority rule, the resulting final decoded output would still be correct.

For this study, five gating and movement networks were trained for each subpopulation of neurons, and the top three networks were selected to form the committee. Although further increases in decoding accuracy can be expected if the committee size were to be increased [26], [28], our choice was based on a trade-off between additional training time and moderate gains in accuracy.

B. *Towards a BMI for Dexterous Control*

The present work is an important step towards the development of a BMI for dexterous control of finger and wrist movements. These results show that it is indeed possible to asynchronously decode individuated and combined movements of the fingers and wrist using randomly selected neuronal ensembles in the M1 hand area. It is important to note, however, that we used sequentially recorded single-unit activities from task-related neurons, and not simultaneously recorded neuronal activity as would be obtained using implanted microelectrode arrays such as the Utah Array [29].

In such an implanted array, all electrodes would not record exclusively from task-related neurons. Therefore, it is possible that activity from non-task-related neurons would hinder the ability of our algorithms to decode the correct movement type. On the other hand, there may also be additional encoded information, due to coherence or phase synchrony within neuron ensembles [30] active during a single task, which could be used to improve the decoding accuracy of complex tasks.

Recordings from a microelectrode array also would be spatially constrained to the recording volume of the array. The lack of somatotopic organization in the M1 hand area, however, offers the possibility of successful decoding irrespective of the precise recording location selected. Our preliminary work [31] using constrained virtual neuronal ensembles that mimic the reality of recording from actual electrode arrays placed in the target area, show comparable decoding accuracy to the results in this paper. Current work [32] also investigates how decoding performance is affected by constraining these virtual neuronal ensembles to volumes that correspond to actual electrode array

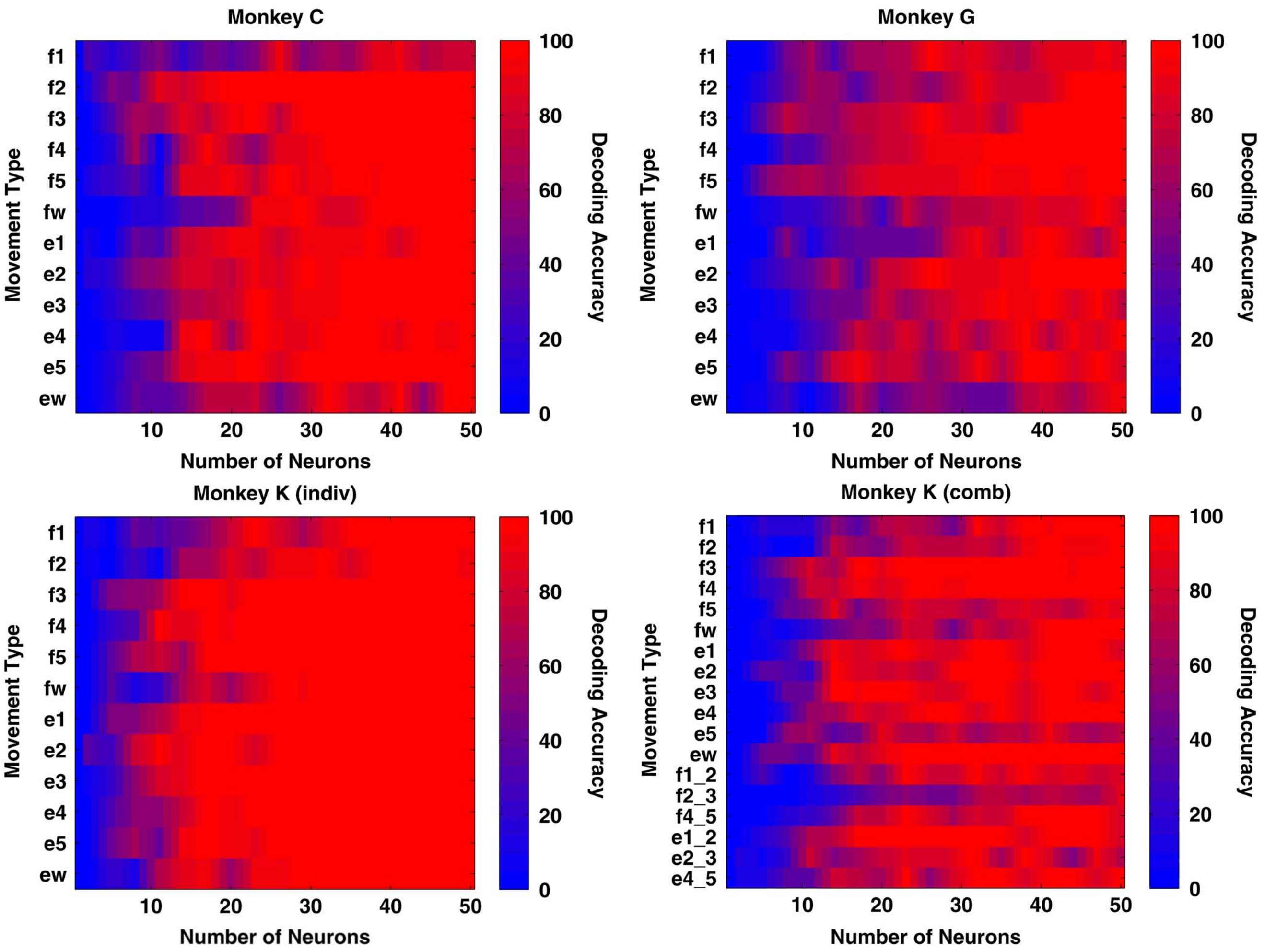

Fig. 9. Asynchronous decoding results for each movement type. Ten out of 12 individuated movement types are decoded with $> 95\%$ accuracy for monkey C, and all 12 individuated movement types are decoded with $> 99\%$ accuracy for monkey K using 50 neurons. When combined movements of two digits are included, 13 out 18 movement types are decoded with $> 99\%$ for monkey K using 50 neurons.

architecture dimensions. This will be invaluable in determining both the optimal location for implantation in the M1 hand area, and optimal electrode array sizes required for chronic recording.

Furthermore, it is to be noted that our decoding algorithm currently only allows for decoding of discrete states that have been previously presented to the classifier. However, for a practical BMI application a further desirable trait would be the ability to predict states that have not have been presented to the classifier during training (i.e., decoding complex dexterous movements such as flexion/extension of multiple fingers or complex grasps using classifiers trained only on dexterous primitives such as flexion/extension of individual fingers). Preliminary work suggests that our algorithms are easily adaptable to decode actions of two fingers simultaneously having only trained on the movement of individual fingers. This will be of particular importance for a BMI where it is impractical to train the classifier for every possible motor action. Lastly, although the experimental paradigm for the current study was a discrete flexion/extension task, it is desirable for future studies to incorporate graded movements where continuous state decoding for varying degrees of flexions and extensions can be achieved.

While actual closed-loop use of a BMI will involve some degree of operant conditioning, the native decoding achieved here suggests that such retraining might be minimal, providing relatively intuitive control of the individual fingers in a prosthetic hand. Although closed-loop experiments with additional monkeys utilizing implanted arrays can extend the present findings, this work opens the door for developing a direct neural interface to provide real-time, dexterous control of a next-generation multifingered hand neuroprosthesis.

REFERENCES

[1] G. Santhanam, S. I. Ryu, B. M. Yu, A. Afshar, and K. V. Shenoy, "A high-performance brain-computer interface," *Nature*, vol. 442, pp. 195–8, Jul. 13, 2006.

[2] J. Wessberg, C. R. Stambaugh, J. D. Kralik, P. D. Beck, M. Laubach, J. K. Chapin, J. Kim, S. J. Biggs, M. A. Srinivasan, and M. A. Nicolelis, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature*, vol. 408, pp. 361–5, Nov. 16, 2000.

[3] J. K. Chapin, K. A. Moxon, R. S. Markowitz, and M. A. Nicolelis, "Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex," *Nat. Neurosci.*, vol. 2, pp. 664–70, Jul. 1999.

[4] E. E. Fetz, "Real-time control of a robotic arm by neuronal ensembles," *Nat. Neurosci.*, vol. 2, pp. 583–4, Jul. 1999.

[5] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. A. 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, pp. 164–71, Jul. 13, 2006.

[6] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows, and J. P. Donoghue, "Instant neural control of a movement signal," *Nature*, vol. 416, pp. 141–2, Mar. 14, 2002.

[7] S. Musallam, B. D. Corneil, B. Greger, H. Scherberger, and R. A. Andersen, "Cognitive control signals for neural prosthetics," *Science*, vol. 305, pp. 258–62, Jul. 9, 2004.

[8] L. H. Snyder, A. P. Batista, and R. A. Andersen, "Coding of intention in the posterior parietal cortex," *Nature*, vol. 386, pp. 167–70, Mar. 13, 1997.

[9] D. M. Taylor, S. I. Tillery, and A. B. Schwartz, "Direct cortical control of 3-D neuroprosthetic devices," *Science*, vol. 296, pp. 1829–32, Jun. 7, 2002.

[10] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. Nicolelis, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS Biol.*, vol. 1, p. E42, Nov. 2003.

[11] A. V. Poliakov and M. H. Schieber, "Limited functional grouping of neurons in the motor cortex hand area during individuated finger movements: A cluster analysis," *J. Neurophysiol.*, vol. 82, pp. 3488–505, Dec. 1999.

[12] M. H. Schieber, "Individuated finger movements of rhesus monkeys: A means of quantifying the independence of the digits," *J. Neurophysiol.*, vol. 65, pp. 1381–91, Jun. 1991.

[13] M. H. Schieber, "Motor cortex and the distributed anatomy of finger movements," *Adv. Exp. Med. Biol.*, vol. 508, pp. 411–6, 2002.

[14] M. H. Schieber and L. S. Hibbard, "How somatotopic is the motor cortex hand area?," *Science*, vol. 261, pp. 489–92, Jul. 23, 1993.

[15] A. P. Georgopoulos, G. Pellizzer, A. V. Poliakov, and M. H. Schieber, "Neural coding of finger and wrist movements," *J. Comput. Neurosci.*, vol. 6, pp. 279–88, May–Jun. 1999.

[16] P. Dechent and J. Frahm, "Functional somatotopy of finger representations in human primary motor cortex," *Human Brain Mapp.*, vol. 18, pp. 272–83, Apr. 2003.

[17] I. Indovina and J. N. Sanes, "On somatotopic representation centers for finger movements in human primary motor cortex and supplementary motor area," *NeuroImage*, vol. 13, pp. 1027–34, Jun. 2001.

[18] A. P. Georgopoulos, A. B. Schwartz, and R. E. Kettner, "Neuronal population coding of movement direction," *Science*, vol. 233, pp. 1416–9, Sep. 26, 1986.

[19] S. B. Hamed, M. H. Schieber, and A. Pouget, "Decoding m1 neurons during multiple finger movements," *J Neurophysiol*, vol. 98, pp. 327–33, Jul. 2007.

[20] J. Wessberg and M. A. Nicolelis, "Optimizing a linear algorithm for real-time robotic control using chronic cortical ensemble recordings in monkeys," *J. Cogn. Neurosci.*, vol. 16, pp. 1022–35, Jul.–Aug. 2004.

[21] W. Wu, Y. Gao, E. Bienenstock, J. P. Donoghue, and M. J. Black, "Bayesian population decoding of motor cortical activity using a kalman filter," *Neural Comput.*, vol. 18, pp. 80–118, Jan. 2006.

[22] W. Wu, A. Shaikhouni, J. Donoghue, and M. Black, "Closed-loop neural control of cursor motion using a kalman filter," in *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, 2004, vol. 6, pp. 4126–9.

[23] C. Kemere, K. V. Shenoy, and T. H. Meng, "Model-based neural decoding of reaching movements: A maximum likelihood approach," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 925–32, Jun. 2004.

[24] K. V. Shenoy, D. Meeker, S. Cao, S. A. Kureshi, B. Pesaran, C. A. Buneo, A. P. Batista, P. P. Mitra, J. W. Burdick, and R. A. Andersen, "Neural prosthetic control signals from plan activity," *Neuroreport*, vol. 14, pp. 591–6, Mar. 24, 2003.

[25] C. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford Univ. Press, 1995.

[26] A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation, and active learning," *Adv. Neural Inf. Process. Syst.*, vol. 7, pp. 231–238, 1995.

[27] M. Moller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks*, vol. 6, pp. 525–533, 1993.

[28] R. Polikar, "Ensemble based systems in decision making," *IEEE Cir. Syst. Mag.*, vol. 6, no. 3, pp. 21–45, 2006.

[29] E. M. Maynard, C. T. Nordhausen, and R. A. Normann, "The utah intracortical electrode array: A recording structure for potential brain-computer interfaces," *Electroencephalogr .Clin. Neurophysiol.*, vol. 102, pp. 228–39, Mar. 1997.

[30] B. B. Averbeck and D. Lee, "Coding and transmission of information by neural ensembles," *Trends Neurosci.*, vol. 27, pp. 225–30, Apr. 2004.

[31] S. Acharya, V. Aggarwal, F. Tenore, H. C. Shin, R. Etienne-Cummings, M. H. Schieber, and N. V. Thakor, "Towards a brain-computer interface for dexterous control of a multi-fingered prosthetic hand," in *3rd Int. IEEE EMBS Conf. Neural Eng.*, Kohala Coast, HI, May 2007, pp. 200–203.

[32] S. Acharya, F. Tenore, V. Aggarwal, R. Etienne-Cummings, M. H. Schieber, and N. V. Thakor, "Decoding finger movements using volume-constrained neuronal ensembles in M1 hand area," *IEEE Trans. Neural. Syst. Rehabil. Eng.*, 2007, submitted for publication.

Vikram Aggarwal (S'05) received the B.ASc. degree in computer engineering from the University of Waterloo, Canada, in 2005, and the M.S.E. degree in biomedical engineering, in 2007, from Johns Hopkins University, Baltimore, MD, where he is currently working towards the Ph.D. degree in biomedical engineering.

His research interests include brain–computer interfaces, neuroprosthetics, and biomedical instrumentation.

Mr. Aggarwal is a student member of the Biomedical Engineering Society. He was an IEEE Student Paper Night winner ('05), IEEE EMBS Student Design Competition finalist at IEEE EMBS ('06), and student fellowship recipient at IEEE EMBS on Neural Engineering (NER '07). He is currently a Medtronic Scholar.

Soumyadipta Acharya (S'05) received the M.B.B.S. degree from Calcutta University, India, in 2000 and served as Resident House Physician in Cardiology at SSKM Hospital, Calcutta, India. He received the M.S.E. degree in biomedical engineering from the University of Akron, OH, in 2004. He is currently working toward the Ph.D. degree in biomedical engineering at the Johns Hopkins University, Baltimore, MD. His primary research interests are neural signal processing, brain–machine interfaces, and neuroprosthetics.

Dr. Acharya has been the recipient of the Outstanding Graduate Student in BME Award ('04), University of Akron President's Letter of Commendation for Excellence in Research ('04) NASA Tech Briefs Award ('06), and the Outstanding Teaching Assistant Award ('07).

Francesco Tenore (S'97–M'06) received the Italian laurea degree in electronics engineering from the University of Trieste, Trieste, Italy, in 2000, and the M.S.E. degree in electrical engineering, in 2002, from the Johns Hopkins University, Baltimore, MD, where he is currently working toward the Ph.D. degree in electrical engineering.

His research focuses on neuromorphic circuits and systems for control of upper and lower limb prostheses, and has been featured in scientific divulgation channels such as Beyond Tomorrow (Australia, 2005), Technology Review (2006) and EE Times (2007).

Mr. Tenore was a recipient of the National Institutes of Health Neuroengineering Training grant in 2006.

Vikram Aggarwal (S'05) received the B.ASc. degree in computer engineering from the University of Waterloo, Canada, in 2005, and the M.S.E. degree in biomedical engineering, in 2007, from Johns Hopkins University, Baltimore, MD, where he is currently working towards the Ph.D. degree in biomedical engineering.

His research interests include brain–computer interfaces, neuroprosthetics, and biomedical instrumentation.

Mr. Aggarwal is a student member of the Biomedical Engineering Society. He was an IEEE Student Paper Night winner ('05), IEEE EMBS Student Design Competition finalist at IEEE EMBS ('06), and student fellowship recipient at IEEE EMBS on Neural Engineering (NER '07). He is currently a Medtronic Scholar.

Ralph Etienne-Cummings (M'94–SM'98) received the B.Sc. degree in physics from Lincoln University of Pennsylvania, in 1988, and the M.S.E.E. and Ph.D. degrees in electrical engineering from the University of Pennsylvania, Philadelphia, in 1991 and 1994, respectively.

Currently, he is an Associate Professor of computer engineering at The Johns Hopkins University (JHU), Baltimore, MD. He is the Director of Computer Engineering at JHU and the Institute of Neuromorphic Engineering. His research interests include mixed-signal VLSI systems, computational sensors, computer vision, neuromorphic engineering, smart structures, mobile robotics and legged locomotion.

Hyun-Chool Shin received the B.Sc., M.Sc., and Ph.D. degrees in electronic and electrical engineering from Pohang University of Science and Technology (POSTECH), Korea, in 1997, 1999, and 2004, respectively.

From 2004 to 2007, he was a postdoctoral researcher at the Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD. He is now a full-time lecturer in the School of Engineering, College of Information and Technology, Soongsil University, Korea. His research interests include adaptive filtering theory for wireless communications, laser speckle imaging for brain microvessel, and bio/neurological signal processing such as electroencephalogram, evoked potential, and neural spike analysis.

Marc H. Schieber received the A.B. degree in 1974, and the M.D. and Ph.D. degrees, in 1982, all from Washington University, St. Louis, MO.

He currently is Professor of Neurology and of Neurobiology and Anatomy at the University of Rochester, and Attending Neurologist on the Brain Injury Rehabilitation Unit at St. Mary's Hospital, Rochester, NY. His research focuses on how the nervous system controls muscles to perform dexterous movements of the hand and fingers.

Dr. Schieber is a member of the Society for Neural Control of Movement and Society for Neuroscience. He currently is recipient of an NINDS Javits Investigator Merit Award.

Nitish V. Thakor (F'97) received the B.Tech. degree in electrical engineering from Indian Institute of Technology, Bombay, in 1974 and the Ph.D. degree in electrical and computer engineering from the University of Wisconsin, Madison, in 1981.

He is currently a Professor of biomedical engineering at the Johns Hopkins University, Baltimore, MD. He conducts research on neurological instrumentation, biomedical signal processing, micro and nanotechnologies, neural prosthesis, and clinical applications of neural and rehabilitation technologies. He has authored more than 170 peer-reviewed publications on these subjects.

Dr. Thakor is Editor-in-Chief of the IEEE TRANSACTIONS ON NEURAL AND REHABILITATION ENGINEERING, Director of the NIH Training Grant on Neuroengineering, a Fellow of the American Institute of Medical and Biological Engineering, and Founding Fellow of the Biomedical Engineering Society. He is a recipient of a Research Career Development Award from the National Institutes of Health and a Presidential Young Investigator Award from the National Science Foundation. He is also a recipient of the Centennial Medal from the University of Wisconsin School of Engineering, Honorary Membership from Alpha Eta Mu Beta Biomedical Engineering Student Honor Society and Distinguished Service Award from IIT Bombay.