

# NIH Public Access

Author Manuscript

Biomed Signal Process Control. Author manuscript; available in PMC 2013 November 01

# Published in final edited form as:

Biomed Signal Process Control. 2012 November 1; 7(6): 632–639. doi:10.1016/j.bspc.2012.03.003.

# Neuron Selection by Relative Importance for Neural Decoding of Dexterous Finger Prosthesis Control Application

# Hyoung-Nam Kim,

Department of Electronics Engineering, Pusan National University, Busan 609-735, Korea. Department of Biomedical Engineering, Johns Hopkins School of Medicine, Baltimore, MD 21205 USA

# Yong-Hee Kim,

Department of Electronics Engineering, Pusan National University, Busan 609-735, Korea

# Hyun-Chool Shin,

School of Electronics Engineering, Soongsil University, Seoul 156-734, Korea

# Vikram Aggarwal,

Department of Biomedical Engineering, Johns Hopkins School of Medicine, Baltimore, MD 21205 USA

# Marc H. Schieber, and

Department of Neurology, Neurobiology and Anatomy, Brain and Cognitive Sciences and Physical Medicine and Rehabilitation, University of Rochester Medical Center, Rochester, NY 14642 USA

# Nitish V. Thakor

Department of Biomedical Engineering, Johns Hopkins School of Medicine, Baltimore, MD 21205 USA

Hyoung-Nam Kim: hnkim@pusan.ac.kr; Nitish V. Thakor: nitish@jhu.edu

# Abstract

Future generations of upper limb prosthesis will have dexterous hand with individual fingers and will be controlled directly by neural signals. Neurons from the primary motor (M1) cortex code for finger movements and provide the source for neural control of dexterous prosthesis. Each neuron's activation can be quantified by the change in firing rate before and after finger movement, and the quantified value is then represented by the neural activity over each trial for the intended movement. Since this neural activity varies with the intended movement, we define the relative importance of each neuron independent of specific intended movements. The relative importance of each neuron is determined by the inter-movement variance of the neural activities for respective intended movements. Neurons are ranked by the relative importance and then a subpopulation of rank-ordered neurons is selected for the neural decoding. The use of the proposed neuron selection method in individual finger movements improved decoding accuracy by 21.5% in the case of decoding with only 5 neurons and by 9.2% in the case of decoding with only 15 highly-ranked neurons, a decoding accuracy of 99.5% was

Correspondence to: Hyoung-Nam Kim, hnkim@pusan.ac.krhkim146@jhmi.edu; Yong-Hee Kim.

<sup>© 2012</sup> Elsevier Ltd. All rights reserved.

**Publisher's Disclaimer:** This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

achieved. The performance improvement is still maintained when combined movements of two fingers were included though the decoding accuracy fell to 95.7%. Since the proposed neuron selection method can achieve the targeting accuracy of decoding algorithms with less number of input neurons, it can be significant for developing brain-machine interfaces for direct neural control of hand prostheses.

# Index Terms

Brain-Machine Interface (BMI); neural decoding; relative importance; neuron selection

# 1. INTRODUCTION

A Brain-Machine Interface (BMI) is a methodology which enables a brain to communicate with an external device bypassing normal neuromuscular systems. Currently, a BMI has drawn much interest as an appropriate alternative for restoring both motor control [1–4] and sensory feedback to amputees so that they can again perceive heat, cold, pressure, and the position of a limb in space [1]. BMI systems collect neural activities from various cortical areas, such as the primary motor, premotor and posterior parietal cortex, and interpret the encoded motor-intent into control commands or kinematic parameters [2–4]. Up to now, many relevant studies have been exploited such as a closed-loop control of a computer cursor and target tracking, reaching and grasping task of a hand [5–7].

Presently dexterous, multi fingered prosthetic limbs are under development. Neural control of dexterous hands will require signals from a population of neurons coding for the hand and finger movements. Hence, an important problem in neural prosthesis control is to select, and preferably rank in terms of their relative importance, neurons coding for individual finger movements. Solution to this problem requires a trade-offs between achieving high decoding accuracy and low computational complexity. Since not all recorded neurons contribute equally to the all movements, since some neurons are related weakly or not at all to the specific movements, the use of as many neurons as possible does not guarantee high decoding accuracy and may even degrade the performance of the decoding algorithm. In addition, the increase in the number of input neurons puts a computational burden on finding an optimal solution especially when the goal is to implement such decoding algorithms in an experimental hardware [8, 9]. Therefore, developing a metric for evaluating the contribution of neurons selected for BMI tasks is at the core of designing an efficient real-time BMI. Researchers have developed some techniques to evaluate the relative importance and select the best neurons coding for the information [9-12]. Sensitivity analysis and single neuron correlation analysis through a vector linear model were proposed to quantitatively rate the importance of neurons in neural to motor mapping [9, 10]. These analyses depend on the decoding model and thus they are not easy to interpret from neurophysiologic points of view [10]. In another approach, a neuron's individual removal error was defined and then used for representing its importance in the population vector neural decoding method [11]. Recently, information theoretical analysis based on an instantaneous tuning model was applied to extract the important neuron subset for neural decoding on BMI [12]. However, these quantification methods of neurons' importance have been developed for predicting intended reach and cursor control and are not yet targeted towards achieving dexterous hand and finger control. Thus, a very active area of research currently is to develop neural control of dexterous hand prosthesis, i.e. provide realistic control strategies for actuation and control of individual and combined finger movements [13–23]. However, methods of neuron selection for complex finger movements, have not been well developed, and as such is the subject of this paper.

We present a new simple metric for quantifying the relative contribution of a neuron towards finger movements. Based on the change in firing rate before and after the starting moment of each instructed finger movement, we define a random variable for the change in firing rate. Then, finding the means of each random variable over six trials for each movement, we finally compute the variance of the means over whole movements for each neuron. A larger variance of neural activations over each finger movement, and it can contribute to accurate decoding performance of all finger movements. Thus, we use these variances as a new metric for ordering neurons. With the ordered neurons we performed maximum-likelihood (ML) neural decoding [20] and then compared the performance with that of randomly selected neurons. Our objective is to demonstrate an improvement in decoding accuracy.

The remainder of the paper is organized as follows: In Section II, neuron selection based on neural activity is developed and ML neural decoding with the selected neurons is introduced. Section III shows the performance improvements with the selected neurons by comparing ML decoding performances with and without selected neurons. We analyze the decoding performance when both individual and combined finger movements are included. Section IV presents the conclusion.

# 2. Neural Decoding Based on Neuron Selection of M1 Neurons

# 2.1. Neuronal Recordings from Motor (M1) Cortex

Three male rhesus (Macaca mulatta) monkeys—K, G, C—were trained to perform visuallycued movements of individual fingers and the wrist movements. In addition, the monkey K was trained to perform combined finger movements involving two digits in order to test the decoding accuracy of more dexterous and complex movements. The monkeys were 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 [18]. These recording were obtained using self-made, glasscoated, Pt-Ir, microelectrodes. Recording tips were etched to be parabolic in shape and approximately 10  $\mu m$  wide 20  $\mu m$  back from the tip [21]. There were 12 distinct individual movements: flexion (f) and extension (e) of each of the fingers (1=thumb, ..., 5=little), the wrist (w) of the right hand, and six combined two-finger movements:  $f_{12}$ ,  $f_{23}$ ,  $f_{45}$ ,  $e_{12}$ ,  $e_{23}$ , e45. The monkeys placed their right hand in a pistol-grip manipulandum; this grip separated each finger into a different slot. The pistol grip manipulandum was also mounted on an axis allowing flexion and extension of the wrist. The monkeys were instructed to flex or extend a single digit until a microswitch was closed. The duration of each trial was approximately 2 sec, and for analysis all trials were aligned such that switch closure occurred at 1 sec [18]. Throughout these investigations, the monkeys were cared for according to the "Guiding Principles for Research Involving Animals and Human Beings" accepted by the American Physiological Society [13]. A detailed description of the methods used to train the monkey and the actual experimental protocols can be found in [12] and [13]. Single-unit activities were recorded from 115 task-related neurons in the M1 cortex of the monkey. Independent trials of each type of movements were recorded six times.

# 2.2. Ordering of Neurons by Relative Importance

To define the neural activity, we need a random variable representing firing rate of a neuron for each finger movement. Let  $r_n(m)$  be a random variable of firing rate of a neuron *n* for a movement type of *m*. Specifically,  $r_n(0_m)$  denotes the baseline activity of the neuron *n* before the movement of *m*. Then, we can define the neural activity considering only movement by introducing the following random variable [20]:

$$x_n(m) = r_n(m) - r_n(0_m).$$
 (1)

Since the random variable of  $x_n(m)$  represents the change in firing rate before and after the starting moment of instructed finger movement, it can be used as a metric of neuron's activation for respective movements. Considering the randomness of neural activity, we can determine the neuron's sensitivity to a particular finger movement *m* by obtaining the ensemble average of  $x_n(m)$ , *i.e.*,  $E[x_n(m)]$ . The estimate of  $E[x_n(m)]$  is usually computed by averaging the recorded neural activation,  $x_n(m, k)$ , for possible training sets, that is

$$\mu_n(m) = \frac{1}{P} \sum_{p=1}^{P} x_n(m, p), \quad (2)$$

where *P* is the number of independent training sets. As *P* increases, the reliability of the metric can be also improved. However, the increase of *P* means that more training data are needed and thus there is a trade-off between the data size and the metric reliability. Letting *M* be the total number of tested movement types, then the neuron *n* has  $M\mu_n(m)$ s and each  $\mu_n(m)$  represents the estimate of the neuron's activity corresponding to the movement of *m*. As a result, the value of  $\mu_n(m)$  can be considered a straightforward metric for the absolute degree of neural activation ascertaining how much the neuron *n* contributes to a particular movement of *m* independent of other movements. This metric, however, cannot be directly applied for selecting the input neurons because the goal of neural decoding is to find the unknown movement from recorded spike signals. For any metric to be available for ascertaining the importance of a neuron when selecting an appropriate input neuron set for neural decoding, it should reflect the relative difference of activations among all the tested movements, not the absolute magnitude of activation for a particular movement. To achieve this goal, we define a relative importance of a neuron *n* with the inter-movement variance of neural activities as the following equation,

$$V_n = \frac{1}{M} \sum_{m=1}^{M} (\mu_n(m) - \overline{\mu}_n)^2,$$
 (3)

where  $\mu_n$  is the mean of  $\mu_n(m)$  for the tested movements given by

$$\overline{\mu}_n = \frac{1}{M} \sum_{m=1}^M \mu_n(m). \quad (4)$$

For a neuron to be distinguishably activated for each finger movement, the  $\mu_n(m)$  of a neuron should be separable for each *m* so as to mitigate the probability of false detection in the neural decoder. That is even if the  $\mu_n(m)$  of a neuron is large for several different movements, there could be a case where those movements are analogous to each other, then its contribution for neural decoding becomes relatively low as shown in Fig. 1(a). On the other hand, Fig. 1(b) shows the opposite case where a neuron is slightly activated for most of movements but its neural activations,  $\mu_n(m)$ s, are separable from each other, then this neuron has high impact contribution for neural decoding. Fig. 2 briefly describes the process to calculate the relative importance of each neuron: we compute the estimates of the neural activities,  $\mu_n(m)$  (*m*=1, ..., *M*), for a neuron *n* over each trial data and then find the intermovement variance of the neural activities. Specially, in the ML decoding method, the

neural activities,  $\mu_n(m)$ , represent the estimates of the ensemble means of each likelihood function [20]. After computing the relative importance of each neuron,  $V_n$ , we can rank neurons by  $V_n$  for neural decoding.

#### 2.3. Neural Decoding with Selected Neurons

Neuron selection based on the proposed method of assigning relative importance of each neuron to a given movement is independent of the model used for neural decoding and thus may contribute to reduction of the required number of neurons for achieving the target performance. Several different methods have been presented for decoding finger movements from neural activity [12–20] and we can use any of these to show the effectiveness of our neuron selection method. Specifically, we will use the ML decoding method described in [20] which is based on Skellam and Gaussian distributions.

ML decoding estimates an unknown parameter, *m*, corresponding to the intended movement, so that the probability function  $Pr(x_1, x_2, ..., x_N m)$  is maximized, *i.e.* 

$$\widehat{m} = \arg_m \min \Pr(x_1, x_2, \dots, x_N | m), \quad (5)$$

where *N* is the total number of neurons used for ML decoding. Without any preference of neurons, we may need a large number of neurons to obtain the desired decoding performance because *N* neurons should be randomly selected and thus they might include neurons irrelevant to the movement. If the limiting factor is the total number of neurons, *N*, it is best to choose neurons with high importance for the movement. We choose highly-ranked *N* neurons in the neurons ordered by  $V_n$ .

In order to use (5), we need to define the probability distribution function corresponding to the random variable  $x_n(m)$ . In [20], Skellam and Gaussian distributions were considered and the performance was good for both cases. Skellam and Gaussian distributions are given by

$$s_n(x_n|m) = \alpha_n(m) \left(\frac{\mu_{r_n}(m)}{\mu_{r_n}(0_m)}\right)^{x_n/2} I_x\left(2\sqrt{\mu_{r_n}(m)\mu_{r_n}(0_m)\Delta t^2}\right) \quad (6)$$

and

$$g_n(x_n|m) = \frac{1}{\sqrt{2\pi\sigma_n(m)}} \exp\left(-\frac{(x_n - \mu_n(m))^2}{2\pi\sigma_n^2(m)}\right), \quad (7)$$

where  $\Delta t$  is the time interval for observation of neural spikes,

$$\mu_{r_n(m)} = \frac{1}{P} \sum_{p=1}^{P} r_n(m, p), \quad \mu_{r_n}(0_m) = \frac{1}{P} r_n(0_m, p), \quad (8)$$

 $I_X(z)$  is the modified Bessel function of the first kind,

$$\alpha_n(m) = e^{-(\mu_n(m) + \mu_n(0))\Delta t}, \quad (9)$$

and

$$\sigma_n^2(m) = \frac{1}{P} \sum_{p=1}^{P} (x_n(m, p) - \mu_n(m))^2. \quad (10)$$

Assuming that  $x_i$  and  $x_i$  (*j i*) are independent [24], we get

$$\Pr(x_1, x_2, \dots, x_N | m) = \prod_{n=1}^N s_n(x_n | m) \quad (11)$$

for Skellam distribution and

$$\Pr(x_1, x_2, \dots, x_N | m) = \prod_{m=1}^N g_n(x_n | m) \quad (12)$$

for Gaussian distribution, respectively.

# 3. Results and Discussion

The proposed method for neuron selection was examined by comparing the performance of the ML decoding based on Skellam and Gaussian distribution, using randomly selected neurons and highly-ranked neurons by the proposed method. In order to test the decoding result of various finger movements, we mainly used the spike data recorded from monkey K trained for both individual and combined finger movements. In (1), the firing rate of  $r_n(m)$  was obtained by averaging the number of spikes for 300*ms* after the movement of *m* and the baseline activity of  $r_n(0_m)$  was obtained by averaging the number of spikes for 300*ms* after the movement of *m* and the baseline activity of *r*<sub>n</sub>( $n_m$ ) was obtained by averaging the number of spikes for 800*ms* before the movement of *m*. As shown in Fig. 3, six independent trials were recorded for each movement, with five trials out of them used for training, i.e., *P*=5, and the remaining trial was used for testing. Thus, six different combinations can be used for training and testing in the form of a Jackknife test. In [20], it is shown that the converged performance of the ML decoding based on statistical estimation with five training sets (*P*=5). However, to make a reliable judgment about the validity of the proposed method, we performed the ML decoding of individual finger movements using the spike signals recorded from the other monkeys (G and C).

If *N* neurons are randomly selected among total 115 neurons, there are  $_{I15}C_N$  combinations of input sets. To raise the reliability of decoding performance, 400 input sets out of  $_{I15}C_N$ combinations were randomly selected. On the other hand, for the rank-ordered neurons, if *N* neurons are used for ML decoding, there is only one input neuron set, consisting of the firstranked neuron to the *N*-th ranked one. In order to increase the possible neuron sets, we added next-ranked *L* neurons to the highly-ranked *N* neurons, resulting in (*N*+*L*) ranked neurons. The value of *L* was empirically set to *L*=10 in our testing. Then, a subset of *N* neurons was randomly selected from (*N*+*L*) neurons. So, the possible number of selections becomes  $_{N+L}C_N$  and thus the total number of tested movements for neural decoding was  $12 \times 6 \times_{N+L}C_N$  in the case of 12 finger movements and 6 trials.

# 3.1. Estimate of Neural Activity

We first computed the estimates of the neural activities  $\mu_n(m)$ , which represent the absolute contribution impact of each neuron for the 12 individual movements. This computation was performed to verify capability of the metric for determining neurons' importance. Fig. 4 shows that each neuron has various values of  $\mu_n(m)$  for some different types of movements.

As previously mentioned, each neuron has  $12 \mu_n(m)$ 's and we can get 12 rank-ordered lists as in Table I. To clearly show the effectiveness of  $\mu_n(m)$ , we chose one ordered neuron set out of a list of 12 ranked neurons, which corresponds to one of 12 columns in Table I. Then, we used highly ranked neurons for ML neural decoding and compared the false detection rates over the desired movement, which we denoted as  $m_d$ . Since  $\mu_n(m)$  is most highly related to the movement m, it reflects the lowest false detection rate (FDR) when m equals  $m_d$ . Fig. 5 shows the FDRs for each desired movement, using either 5 (top) or 10 (bottom) neurons ordered by  $\mu_n(1)$ ,  $\mu_n(3)$ ,  $\mu_n(5)$ , as well as the case of a random selection. In the first case, the numbers of 1, 3, and 5 denote the finger movement  $f_1$ ,  $f_3$  and  $f_5$ , respectively. For both N = 5 and N = 10, the FDR was lowest when the desired movement was the same as the movement type whose neural activity was used for ordering neurons. The difference between the lowest FDR and those of the remaining given movement types decreased with 10 ordered neurons.

The ordered result by  $\mu_n(m)$  only shows the activity level of each neuron involved in the particular movement *m*, so it does not assure the accurate neural decoding for the remaining 11 movements. Because the desired movement is not revealed in real environments, the neural activity is not appropriate as a metric for ascertaining the importance of neurons.

# 3.2. Selection of Neurons

The neural activity for the specific movement gives the information mainly related to the specific movement and thus we need to obtain the relative importance by computing the variance of the neural activity,  $V_n$ . For the 12 individual movements, we computed  $V_n$  for each neuron and arranged them in the decreasing order. Fig. 6 shows  $V_n$  and  $\mu_n$  for the ordered neurons. There we can find that some neurons have larger values of  $\mu_n$  than others. Since  $\mu_n$  is the mean value of  $\mu_n(m)$  for all movements, it represents the estimate of the neural activity of the neuron *n*. This means that some neurons with high neural activity may have low relative importance if they have the similar contributions for all movements. From the plot of  $V_n$  we can also find that a knee occurs near the 10<sup>th</sup> neuron, which makes us expect that neural decoding performance would saturate around 10 neurons.

To investigate neural dependency on movement classifications, we also calculated  $V_n$  values for combined two-finger movements and the total movements including both individual and combined movements. To clarify the ordering criterion, let us denote the three variances of the individual, the combined, and the total movements by  $V_{n,L}$ ,  $V_{n,C}$ , and  $V_{n,T}$ , respectively. Figs. 7 and 8 show the values of  $V_{n,C}$  and  $V_{n,T}$  for the combined and the total movements, respectively. It is clear that these three types of movement classifications have different orderings but we need to know how the ordering is changed depending on the movement classifications. From Table II, it can be seen that some neurons have large standard deviations and thus are differently ranked according to movement classifications. For example, neuron 'K20301', which has the largest standard deviation, is ranked 6<sup>th</sup>, 101<sup>th</sup>, and 12<sup>th</sup> for the individual, the combined, and the total movements, respectively. From this analysis we can predict that neural decoding of combined movements with the ordered neurons based on the individual movement does not have comparable performance to that with the ordered neurons by the combined movements.

# 3.3. Differences in Decoding Accuracy According to Selection of Neurons

The effect of neuron selection is evaluated by the ML decoding performance with ordered neurons and with randomly selected neurons. Fig. 9 shows the improved decoding performance by incorporating the proposed neuron selection methods. The criterion for the neuron selection was the variance of the individual movement activity,  $V_{n,F}$  As predicted in the previous section, the decoding accuracy with only 5 neurons approaches 86.8% while the

Page 8

use of randomly selected neurons yields an accuracy of 62.3% with the same number of neurons, and it shows a performance improvement of decoding accuracy by 24.5%. In case of using 10 neurons, the decoding accuracy is also improved by 9.2% compared to the case of randomly-selected neurons. Also, we know that the proposed neuron selection method is robust to decoding algorithms because the performance improvement using Gaussian distribution is similar to that of Skellam distribution. In addition, Fig. 9 shows the better decoding performance than the previous selection method using the tuning depth of a neuron, which was defined as the difference between the maximum and minimum values in the cellular tuning [10]. In addition, we performed the ML decoding using the spike signals recorded from the other subjects (monkey G and C) to generalize the validity of the proposed method. In both cases, the decoding results using the selected neurons by the proposed method show better decoding accuracies compared to the case of using randomly selected neurons as shown in Fig. 10.

We also checked the possibility of using the estimate of the neural activity for neuron selection. As addressed in the previous section, each estimate depends on the corresponding movement type and thus we have to choose one of 12 estimates of  $\mu_n(m)$  which are related to the 12 individual movements, respectively. Fig. 9(a) shows the ML decoding performance of individual movements using different neuron groups chosen by different criteria. As discussed in section III. A, the use of  $\mu_n(m)$  does not give any remarkable merit for neural decoding because it is the partial information only corresponding to the specific movement.

Next, we tried to test the availability of the neuron selection method for decoding of the six combined two-finger movements. Fig. 11 shows the ML decoding performance of combined movements using the different neuron selection methods. For decoding combined movements, the neuron-selection based on the relative importance of the combined movement activity produced an outstanding improvement in decoding performance of 98.3% with 10 neurons. On the other hand, neuron-selection based on the relative importance of individual movements in combined movements does not perform as well, as it yielded an accuracy of 93.5% with 10 neurons. This means that, in order to achieve the improved performance, we need training data for obtaining the required parameters.

Finally we tested the neuron selection methods for total movements. Fig. 12 shows the ML decoding performance of total movements using ordered neurons or not. In this case, the decoding performance does not converge to 100% for any of the neuron selection methods, not even when the number of neurons was increased. Because the usual grasping or control of an object is synthesized by the combination of many individual finger movements, the accurate neural decoding of total finger movements is essential to implement more practical BMI applications and requires further analysis.

## 3. 4. Neural Activity for Multi-Finger and Single-Finger Movements

The average decoding accuracy approaches 99.5% with only 15 neurons when individual movements (Fig. 9(a)) and combined-finger movements (Fig. 11) are separately decoded. However when decoding total movements, the decoding accuracy saturates at 95.7% even with 40 neurons. This phenomenon insinuates that there may be some interrelations between the individual and combined finger movements which confuse the decoder when decoding total movements.

To investigate this problem, we analyzed the main cause of false detections. Table III summarizes the total movement decoding results for each of the individual and combined two-finger movements using 40 neurons. False detections happened mostly in decoding the combined-finger movement tasks of  $f_{23}$ ,  $f_{45}$ ,  $e_{45}$  and the individual movement tasks of  $f_5$ ,  $e_2$ , e5. The detection-rate table shows why the two-finger flexion task of ring and little fingers

has the poorest performance. The false detection falls mainly into the little finger flexion,  $f_5$ , which shows that control independence of finger movements affects brain activity. When the subjects intend to flex only a little finger, the ring finger also flinches irrespective of our intention. Therefore, the flexion of the little finger might have similar neural activity to those of both ring and little fingers. Table III verifies our conjecture because the most false detections for  $f_5$  appeared at  $f_{45}$ . In the same way, we can explain the poor decoding performance for the cases of the two-finger extension task of ring and little fingers,  $e_{45}$  and of the individual extension task of the little finger corresponding to  $e_2$ . In that case, the middle finger also flinches irrespective of monkey's intention. This makes the decoder confused in finding the correct solution between  $e_2$  and  $e_{23}$ . As a result, the poor decoding accuracy for the total finger movements is likely caused by confusion between some individual movements and combined two-finger movements.

# 4. Conclusion

We have presented a neuron selection method by defining the relative importance of each neuron contributing to motor movements. Neuron selection improved the neural decoding performance remarkably due to the use of highly-tuned neurons. Furthermore, neuron selection may decrease the required number of neurons for neural decoding and as a result the number of electrodes needed and the computational power to decode neural activity for achieving dexterous finger actuation in a neuroprosthetic system. By analyzing the improved decoding performance of single- and multi-finger movements, we have taken a step towards the strategy of neural control of dexterous multi-fingered hand.

# Acknowledgments

This research was supported in part by the Korea Research Foundation Grant KRF-2009-013-D00090 (to H-NK) and in part by grants ECCS-0835554 from the National Science Foundation (to NVT), and 1R01EB010100-01 from the National Institutes of Health (to NVT).

# References

- 1. Kennedy PR, Bakay RAE, Moore MM, Adams K, Goldwaithe J. Direct Control of a Computer from the Human Central Nervous System. IEEE Trans Rehab Eng. Jun.2000 8:198–202.
- Wessberg J, Stambaugh CR, Kralik JD, Beck Pamela D, Laubach M, Cahpin JK, Kim Jung, James Biggs S, Srinivasan MA, Nicolelis MAL. Real-time Prediction of Hand Trajectory by Ensembles of Cortical Neurons in Primates. Nature. Nov; 2000 408(6810):361–365. [PubMed: 11099043]
- Moran DW, Schwartz AB. Motor cortical activity during drawing movements: Population Representation During Spiral Tracing. J Neurophysiol. Nov.1999 82:2693–2704. [PubMed: 10561438]
- Taylor DM, Tillery SIH, Schwartz AB. Direct Cortical Control of 3D Neuroprosthetic Devices. Science. Jun.2002 296:1829–1832. [PubMed: 12052948]
- Serruya MD, Hatsopoulos NG, Paninski L, Fellows MR, Donoghue JP. Instant Neural Control of a Movement Signal. Nature. Mar; 2002 416(6877):141–142. [PubMed: 11894084]
- Carmena JM, Lebedev MA, Crist RE, O'Doherty JE, Santucci DM, Dimitrov DF, Patil PG, Henriquez CS, Nicolelis MA. Learning to Control a Brain-machine Interface for Reaching and Grasping by Primates. PLoS Biology. Nov.2003 1(2):article E42.
- Georgopopoulous A, Kalaska J, Caminiti R, Massey J. On the Relations between the Direction of Two-dimensional Arm Movements and Cell Discharge in Primate Motor Cortex. J Nerurosci. May. 1982 2:1527–1537.
- Westwick DT, Pohlmeyr EA, Solla SA, Miller LE, Perreault EJ. Identification of Multiple-input Systems with Highly Coupled Inputs: Application to EMG Prediction from Multiple Intracortical Electrodes. Nerual Computation. Feb; 2006 18(2):329–355.

- Singhal G, Aggarwal V, Acharya S, Aguayo J, He J, Tahkor N. Ensemble Fractional Sensitivity : A Quantitative Approach to Neuron Selection for Decoding Motor Tasks. Comput Intell Neurosci. Feb.2010
- Sanchez JC, Carmerna JM, Lebedev MA, Nicolelis MA, Harris JG, Principe JC. Ascertaining the Importance of Neurons to Develop Better Brain-machine Interfaces. IEEE Trans Biomed Eng. Jun. 2004 51:943–953. [PubMed: 15188862]
- Wahnoun R, He J, Helms Tillery SI. Selection and Parameterization of Cortical Neurons for Neuroprosthetic Control. J Neural Eng. Jun.2006 3:162–171. [PubMed: 16705272]
- Wang Y, Principe Jose C, Sanchez Justin C. Ascertaining Neuron Importance by Information Theoretical Analysis in Motor Brain-Machine Interfaces. Neural Netw. Jul-Aug;2009 22:781–790. [PubMed: 19615852]
- Schieber MH. Individual Finger Novements of Rhesus Monkeys: A Mmeans of Quantifying the Independence of the Digits. J Neurophysiol. Jun; 1991 65(6):1381–1391. [PubMed: 1875247]
- Shieber MH, Hibbard LS. How Somatotopic is the Motor Cortex Hand Area? Science. Jul. 1993 261:489–492. [PubMed: 8332915]
- Poliakov AV, Schieber MH. Limited Functional Grouping of Neurons in the Motor Cortex and Hand Area During Individual Finger Movements: A Cluster Analysis. J Neurophysiol. Dec.1999 82:3488–3505. [PubMed: 10601477]
- Georgopoulos AP, Pellizzer G, Poliakov AV, Schieber MH. Neural Coding of Finger and Wrist Moevments. J Comput Neurosci. May-Jun;1999 6:279–88. [PubMed: 10406138]
- Schieber MH. Motor Cortex and the Distributed Anatomy of Finger Movements. Adv Exp Med Biol. 2002; 508:411–416. [PubMed: 12171137]
- Aggarwal V, Acharya S, Tenore F, Shin H-C, Etienne-Cummings R, Schieber MH, Thakor N. Asynchronous Decoding of Dexterous Finger Movements Using M1 Neurons. IEEE Trans on Neural Syst Rehabil Eng. Feb.2008 16:3–14.
- Acharya S, Tenore F, Aggarwal V, Etienne-Cummings R, Shieber MH, Thakor N. Decoding Individual Finger Movements Using Volume-constrained Neuronal Ensembles in the M1 Hand Area. IEEE Trans Neural Syst Rehabil Eng. Feb.2008 16:15–23. [PubMed: 18303801]
- Shin HC, Aggarwal V, Acharya S, Schieber MH, Thakor N. Neural Decoding of Finger Movements Using Skellam Based Maximum Likelihood Decoding. IEEE Trans Biomed Eng. Mar. 2010 57:754–760. [PubMed: 19403361]
- 21. Aggarwal, V. MS thesis. Johns Hopkins University; Sep. 2007 Deveoloping a Brain-Computer Interface for Control of an Upper-Limb Neuroprosthesis.
- Mollazadeh M, Aggarwal V, Davidson AG, Law AJ, Thakor N, Schieber MH. Spatiotemporal Variation of Multiple Neurophysiological Signals in the Primary Motor Cortex during Dexterous Reach-to-Grasp Movements. J Neurosci. 2011; 31(43):15531–15543. [PubMed: 22031899]
- 23. Aggarwal, V. PhD Dissetation. Johns Hopkins University; 2011. Deveoloping a Brain-Machine Interface for Dexterous Control of an Upper-Limb Neuroprosthesis.
- Jazayeri M, Movshon JA. Optimal Representation of Sensory Information by Neural Populations. Nat Neurosci. May.2006 9:690–696. [PubMed: 16617339]

# Highlights

- We define the relative importance as a variance of averaged neural activation over each stimulus.
- This can be a good measure of the separability of neural activity over each stimulus.
- Decoding accuracy increases if highly-ranked neurons are used for neural decoding.



**Fig. 1.** 

Examples of the probability distributions of the neural activities : (a) the neuron  $n_1$  is highly activated for the movement types, m=1, 2 and 3 but it has similar expected neural activations over each movement, (b) the neuron  $n_2$  has relatively small neural activations for the three movements but they are activated distinguishably.





Two steps for computing the relative importance of each neuron. *P* and *M* denote the number of trials for training and the number of finger movements, respectively.





Measurement of the firing rates before  $(r_n(0_m))$  and after  $(r_n(m=e_2))$  the starting moment of instructed finger movement for a neuron 'K13409'.





Estimates of the neural activities of six randomly-selected neurons recorded from the monkey K for each of the five individual-finger movements.

Kim et al.





False-detection rate for each given movement using 5 neurons (top) and 10 neurons (bottom). Neurons are ordered by  $\mu_n(1)$ ,  $\mu_n(3)$ ,  $\mu_n(5)$  and were randomly-selected.











**Fig. 8.** Variance and mean of  $\mu_n(m)$  of each neuron for total movements.



# Fig. 9.

Performance of the ML decoding methods in individual finger movements: (a) with ordered neurons by the relative importance, the difference between minimum and maximum value, the estimate of the neural activity ( $\mu_n(1)$ ,  $1=f_1$ ), and without any ordering based on Skellam distribution, (b) with ordered neurons by the relative importance, the difference between minimum and maximum value, and without any ordering based on Gaussian distribution.



# Fig. 10.

Performance of the ML decoding methods in individual finger movements with ordered neurons by the relative importance, the estimate of the neural activity ( $\mu_n(1)$ ,  $1=f_1$ ), and without any ordering based on Skellam distribution : (a) in the case of the monkey G, (b) in the case of the monkey C.



# Fig. 11.

Performance of the ML decoding method in combined movements with ordered neurons by the variance of the combined-finger activity, the variance of the individual movement activity, and without any ordering.





Performance of the ML decoding method in total finger movements based on Skellam distribution using ordered neurons by the relative importance and randomly selected neurons.

**NIH-PA** Author Manuscript

of each movement.
$\mu_n(m)$
activity,
e neural
of the
estimate
' each
þ
lists
ordered
l2 rank-
The [

u	-	5	e	4	v	و	7	×	6	10	Ħ	12
rank	$(f_1)$	(f2)	(f3)	( <b>f</b> 4)	$(f_5)$	$(f_w)$	$(e_1)$	(e2)	$(e_3)$	(e4)	(e <sub>5</sub> )	( <i>e</i> <sub>w</sub> )
-	K33504	K21301	K21301	K16301	K21301	K25001	K30833	K30833	K30833	K21301	K30833	K11800
2	K30833	K25001	K11800	K21301	K25001	K16301	K33600	K21101	K14009	K19205	K25001	K13304
3	K11800	K31801	K25001	K11404	K15906	K14009	K21301	K19500	K13407	K14801	K21301	K19500
4	K11707	K22601	K22501	K30833	K30833	K25604	K25001	K22701	K25001	K25001	K14801	K16301
5	K21408	K22701	K22701	K23407	K22701	K21001	K23407	K19205	K11301	K30833	K22406	K21301
9	K11404	K31101	K14009	K25001	K22704	K15904	K11600	K25001	K14801	K16101	K14203	K33605
L	K23701	K14801	K13905	K11800	K22601	K21301	K22701	K22601	K15906	K22501	K22801	K14009
8	K15906	K11404	K16301	K15906	K22801	K14104	K11404	K22704	K11404	K14009	K21201	K22704
6	K21301	K30803	K14001	K23502	K22404	K11404	K15906	K13010	K22704	K11404	K15906	K22701
10	K14801	K11800	K13407	K19101	K19500	K22701	K14203	K14801	K16301	K15906	K11404	K11703

# Table II

Ranked list of neurons for three combinations of movement types.

Rank	Indiv	idual	Combined	Total
1	<u>K30803</u>	Std= 1.1	K33600	<u>K30803</u>
2	K33600	0.5	K33504	K33600
3	K16301	10.1	<u>K30803</u>	K33504
4	K33504	1.0	K14203	K16301
5	K11800	2.8	K31702	K11800
6	<u>K21301</u>	56.2	K35301	K14203
7	K19205	0.0	K19205	K19205
8	K22406	0.0	K22406	K22406
9	K19500	13.3	K11600	K14801
10	K22704	6.0	K11800	K13010
:	:	:	:	:
12	K14801		K23304	<u>K21301</u>
:	:		:	:
101	K13901		<u>K21301</u>	K13208
:	:		:	:

Table III

(N=40).
r movements
l finge
for tota
results
coding
ML de

, m	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_{\rm w}$	e1	e2	e3	е4	65	ew	$f_{12}$	$f_{23}$	$f_{45}$	e 12	623	e 45
рш																		
$f_1$	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$f_2$	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$f_3$	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$f_4$	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$f_5$	0.0	0.0	0.0	0.0	75.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	24.7	0.0	0.0	0.0
$f_{\rm w}$	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$e_1$	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$\mathcal{S}_{1}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	83.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	16.2	0.0
$e_3$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$e_4$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
e,	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	98.1	0.0	0.0	0.0	0.0	0.0	0.0	1.9
$e_{\rm w}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0	0.0
$f_{12}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0	0.0	0.0	0.0
$f_{23}$	0.0	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	97.9	0.0	0.0	0.0	0.0
$f_{45}$	0.0	0.0	0.0	0.0	19.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	80.5	0.0	0.0	0.0
$e_{12}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0	0.0
<b>e</b> 23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100	0.0
$e_{45}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8.8	0.0	0.0	0.0	0.0	0.0	0.0	91.2
<i>m</i> : esti	mated	movem	ent, <i>m</i> c	d: desir	ed move	sment												