Design of A Wearable FMG Sensing System for User Intent Detection during Hand Rehabilitation with a Soft Robotic...

Hong Kai Yap

6th IEEE RAS/EMBS International Conference on
Biomedical Robotics and Biomechatronics (BioRob)
June 26-29, 2016. UTown, Singapore

Design of A Wearable FMG Sensing System for User Intent Detection during Hand Rehabilitation with a Soft Robotic Glove

Hong Kai. Yap[#1, 2], *Student Member, IEEE*, Andrew. Mao[#3], James C.H. Goh[1], and Chen-Hua Yeow*[1, 4], *Member, IEEE*

***Abstract*— This paper presents the design of a wearable feedback system based on force myography (FMG) for user intent detection during hand rehabilitation with a soft robotic glove. We present the development of a form-fitting FMG sensor band using force sensitive resistor (FSR). A supervised learning classifier, Artificial Neural Network (ANN), was implemented to classify four different hand motions with nearly instantaneous prediction speed. Experiments with three healthy subjects were devised to study the training speed and real-time classification accuracy. Results indicate an average training time of less than 95 seconds and a real time accuracy of approximately 95%. The study reveals the successful detection of four different hand motions and a high level of intuitive user intention-based control over the robotic glove.**

I. Introduction

Motor impairment is the most common symptom of neurological disorders, such as stroke, spinal cord injury (SCI) or other neurological trauma. Individuals with impaired motor abilities suffer a restricted ability to perform the activities of daily living (ADL) [1]. Physical therapy for improving hand motor function often involves the repetition of tasks such as grasping or opposition motion [2]. Several robotic devices have been proposed to automate this process [3], among which are hand exoskeleton devices featuring cable-driven, linkage-based or pneumatically driven mechanisms. Recently there has been exploration by several groups into the use of soft robotic actuators for these devices in order to develop devices that are more wearable. Several groups have investigated the use of soft elastomeric actuators for each finger actuated by a pneumatic or hydraulic pump system [4-10]. Unfortunately, while the focus of this research has been on improving the customizability, flexibility and comfort of the glove, the need for user-specific control of the exoskeleton device has largely been ignored. There is therefore a clear need to develop a wearable, customizable control system for rehabilitation based on user intentions.

Much current active research in prosthetics focuses on designing neural interfaces to improve the prediction of upper-extremity postures and movements. A control mechanism that has been the subject of extensive study involves the processing of surface electromyography (sEMG) signals obtained from placing electrodes over various muscle locations on the forearm and upper arm [11]. A more sparsely considered method for control is force myography (FMG) that involves the use of force sensors placed on the surface of the arm to capture expansions or contractions of major muscles groups. A feasibility study of the technique by O. Amft *et al.* yielded encouraging results by successfully distinguishing four types of arm gestures [12]. A more recent study by Xiao *et al.* details the development of a wearable, real-time FMG sensor feedback system for monitoring upper-extremity postures [13].

The aforementioned studies form the foundation of developing a wearable, FMG-based control system to detect user intention. In this paper, we present a complete system for the detection of four different forearm configurations. FMG signals are acquired from 3 muscle locations using force sensitive resistor (FSR) sensors secured by a forearm band with an adjustable Velcro strap. Though proper use of the device necessitates rough localization of the forearm muscle groups, the device was designed to be lightweight and straightforward to use, as well as requiring only a nominal amount of hardware.

In order to develop a real-time control system based on user intentions, a high-performance classifier is needed capable of being implemented on a low-cost microprocessor. In this paper, we outline the use of an artificial neural network (ANN) to classify four different hand configurations, including finger extension, finger flexion, wrist extension and wrist flexion. Though there are many other classifiers that could have been used, among which are the widely investigated Support Vector Machines (SVM) and the Extreme Learning Machine (ELM), ANN was chosen here because of its natural and concise multi-class classification ability and fixed-size parametric model [14], both features which lend itself well to implantation of a mobile, low-cost platform. In this study, both the training and testing phases were implemented on the same control hardware system to both streamline user operation and reduce technical overhead.

This work was supported by the Ministry of Education AcRF Tier 2
*Corresponding author
[#]These authors contributed equally to the work.
[1]Hong Kai Yap, James C. H. Goh and Chen-Hua Yeow are with the Department of Biomedical Engineering, National University of Singapore, 117575 Singapore (e-mail: hongkai@u.nus.edu; biehead@nus.edu.sg; bieych@nus.edu.sg).
[2]Hong Kai Yap is also with the NUS Graduate School for Integrative Sciences and Engineering.
[3]Andrew Mao is with the Department of Biomedical Engineering, Johns Hopkins University, MD 21205 Baltimore, USA. (e-mail: amao@jhu.edu).
[4]Chen-Hua Yeow is also with the Singapore Institute for Neurotechnology and Advanced Robotics Center.

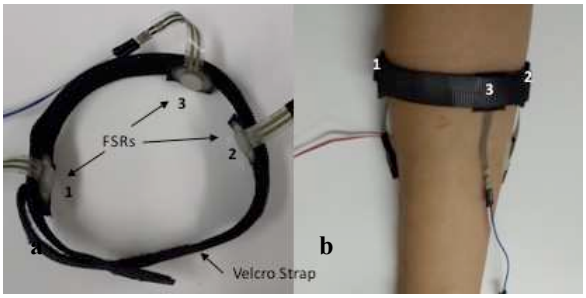

Fig. 1. a) FMG Velcro Strap; b) Sensor Placement on Ventral (Palmar) Aspect of Right Forearm (1: Extensor Carpi Radialis Longus; 2: Flexor Digitorum Superficialis; 3: Flexor Carpi Radialis)

II. METHODS

A. Sensors, Placement and Control Hardware

The FSR sensor is an analog sensor made of a polymer thick film that decreases in resistance when pressure is applied to the sensing area. Three circular FSR sensors (FSR402, Interlink Electronics, USA) were attached to a Velcro band with circular pad attached to the middle to improve the contact with the user's forearm. Each sensor is attached to small Velcro square, allowing the location of the sensors on the band to be readjusted based on the best location for sensing the major forearm muscles, i.e. the flexor digitorum superficialis, the flexor carpi radialis, and the extensor carpi radialis longus. The size of the band is adjustable to a wide range of forearm circumferences. The total mass of the final band with the sensors attached was less than 50g, resulting in a very lightweight device with straightforward operation by the user (Fig. 1).

With the given hardware setup, four user hand motions were found to be readily detectable and were chosen to be mapped to assisted actions on the soft robotic glove. A summary of this is provided in Table 1. The sensors described in the previous section were connected to the 8-bit analog to digital pins on the Arduino Mega 2560, the main control board where signal processing took place. The implemented algorithm required only a low-end processor with small amounts of RAM, making the system well suited for mobile application.

TABLE I. HAND CONFIGURATION AND MAPPED GLOVE ACTIONS

Class Label	Hand Configuration	Mapped Glove Action
1	Finger Extension	All Actuators Off
2	Finger Flexion	Palmar Grasp (Four Actuators On)
3	Wrist Flexion	Index Finger Flexion (One Actuator On)
4	Wrist Extension	Index, Middle Finger Flexion (Two Actuators On)

B. Multilayer Perceptron

In this system, real-time classification on the control board is achieved through the use of an ANN, a popular and widely used class of supervised learning classifiers [3]. Specifically, a multilayer perceptron (MLP) is implemented using a feed-forward back-propagation training model. The topology of the perceptron involves three distinct layers of neurons, or nodes: the input layer, the hidden layer(s), and the output layer. Each node in one layer is connected to every node in the successive layer with varying connection strength. These connections relay output signals from one layer to the next. The number of nodes in the input layer is equal to the dimensionality of the feature vector, whereas the number of classes being investigated determines the number of nodes in the output layer. The number of hidden layers and the number of hidden neurons in each layer depends on the application, but one layer is generally sufficient for most applications, and a single hidden layer has been shown to be a universal approximator [15].

Each node in the neural network uses the sum of the weighted connections with each node in the previous layer to compute its own output according to an activation function. The input to each neuron [16] is defined as:

$$In_i = \sum_j w_{ij} a_j + bias_i \quad (1)$$

where w_{ij} represents the connection strength between node j in the previous layer and the current node i, a_j represents the activation of node j in the previous layer, and $bias_i$ represents the resting state of the neuron that is added to the sum of the other inputs. The bias is important because it increases the range of possible solutions and allows a signal to travel through the network in the absence of input. The result, In_i, is then passed through the nonlinear activation function to compute the output state, a_i.

$$a_i = \frac{1}{1+e^{-In_i}} \quad (2)$$

Like other supervised learning classifiers, the MLP is trained by example. To begin the training, a set of representative input and output patterns are selected. As each pattern is presented to the network, the model "learns" the patterns by adjusting the strengths of the connections in the network to approach the desired output. The back-propagation algorithm uses the gradient descent method to minimize the network error, E, defined as follows:

$$E = \sum_p \sum_i \left(T_{pi} - O_{pi}\right)^2 \quad (3)$$

where p is an index for the training patterns, i is an index for the output nodes, T_{pi} is the target output and O_{pi} is the activation of the output node. By the gradient descent technique, the weight update rule for each node is:

$$\Delta w_{ij}[n+1] = \varepsilon \gamma_{pi} a_{pj} + \alpha \Delta w_{ij}[n] \quad (4)$$

where w_{ij} is the connection strength between nodes i and j, γ_{pi} is the error of node i for pattern p, a_{pj} is the activation of node j for pattern p, ε is the learning rate parameter, and α is the momentum parameter. The learning rate affects how much of the error is back-propagated, and the momentum parameter modulates how much of previous iteration influences the current iteration. Both are network parameters between 0 and 1. The node error for an output node, γ_{pi}, is given by:

$$\gamma_{pi} = \left(T_{pi} - a_{pi}\right) a_{pi} \left(1 - a_{pi}\right) \quad (5)$$

where the first term is the difference between the target pattern and the node's output, and the last two terms represent the derivative of the sigmoid activation function. The error for a hidden node is slightly more complex because there is no target to measure against. Instead, the first term for the hidden node error depends on the relationship between the weights and the error for the output node, given by the equation:

$$\gamma_{pj} = a_{pj}(1 - a_{pj}) \sum_k \gamma_{pi} w_{pi} \quad (6)$$

where the first two terms are the derivative of the activation at the hidden node, and the summation collects the weighted errors from the outer layer. It is thus in this computation that the error is "back-propagated" to the hidden layer. The network is trained until the error is reduced below a certain threshold. More details on the back-propagation algorithm can be found in [16].

C. Network Parameters

As we are interested in classifying 3-channel FMG sensor data into four distinct classes, the number of input nodes was set to three. To reduce the complexity of the model on a control board with limited RAM, the number of output nodes was set to two (a binary output). Though there are general rules of thumb for choosing the network parameters, through a combination of extensive testing and network "pruning" via manual inspection, the classifier was found to train most effectively with 7 hidden nodes, a learning rate ε of 0.10, and a momentum α of 0.9. The condition for termination of training was chosen to be by minimum error. A summary comparison of the chosen model parameters with the typical values is shown in Table 2.

TABLE II. CHOSEN NETWORK PARAMETERS

Parameter	Possible Values	Typical Value	Chosen Value
Number of Hidden Neurons	$[1, \infty] \in \mathbb{Z}$	8	7
Learning Rate ε	$[0, 1] \in \mathbb{R}$	0.3	0.1
Momentum α	$[0, 1] \in \mathbb{R}$	0.9	0.9
Epochs or Training Cycles	$[1, \infty] \in \mathbb{Z}$	5000000	5000000
Termination Condition	Epoch or Min Error	Min Error	Min Error
Minimum Error	$[0, 0.5] \in \mathbb{R}$	0.01	0.004

Unfortunately, the well-known drawbacks to the use of ANNs is the possibility of converging on local minima and not global minima during the training phase, regardless of the number of epochs, and the possibility of over-fitting the data, which creates excessive variance in the model predictions. Theses concern was mitigated slightly by choosing a lower learning rate and by modifying the training protocol to ensure a more representative data set for each grasping configuration, as discussed previously. In this study, we chose to train the model based on three distinct sensor readings for each hand configuration. Though this is much smaller than the standard training data set for a model of this size, it was more important and more practical for this application to obtain a smaller, representative data set for each configuration with a quick training speed than to have twenty of the same training pattern and a slower training speed. In the end, though these drawbacks present a slight concern for the training and generalization ability of the classifier, the relatively fast training and quick prediction time without loss of accuracy makes MLP a good fit for real-time classification, and represents a distinct advantage over the use of SVM.

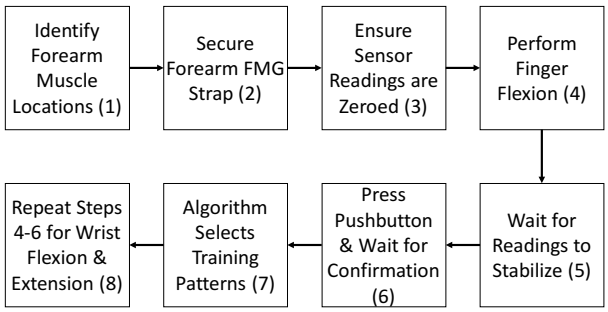

Fig. 2. Training Flowchart.

D. Training and Experimental Protocols

Three healthy volunteer subjects were recruited to investigate the real-time classification accuracy of the system. The program was designed to simplify and automate the training process while combining the testing phase on the same platform, eliminating any intermediate manual processing steps and streamlining system operation. To begin, the user identified the major forearm muscles, as described above, adjusted the sensor locations and secured the band. At the beginning of the training phase, the program continuously output the sensor readings, allowing the user a chance to calibrate the band before proceeding with the training. After the calibration phase, the subject sequentially performed each hand configuration, simply pressing a pushbutton to prompt the program to automatically select training samples. The program would essentially select sequential samples only when the vectorized 3-channel reading was a certain distance away from the previous sample in that configuration (Fig.2). The program would automatically begin training the MLP once the last configuration was confirmed.

During the testing phase, the participant was asked to rest his elbow on the table and perform the hand postures sequentially according to a set of instructions printed to the monitor. The user was first asked to maintain the default position, finger extension, for ten seconds, during which the sensor readings, classification, and time are continuously printed for the user to see. After these ten seconds, the instruction to change to the next grasping configuration, finger flexion, was displayed, with no time allotted for the user to transition between configurations. Therefore, after the prompt was given, the user was expected to directly transition into the next grasping motion. After another ten seconds, another instruction was provided to perform the wrist flexion, and after another ten seconds, to perform the wrist extension. The user was then asked to return to the finger extension configuration, completing one "cycle" of the testing phase where the user had sequentially performed each of the four

possible grasping motions. This cycle was repeated twice more, for a total of three cycles per test.

The network was trained on normalized values of the digitized 3-channel sensor readings. After the analog readings were converted to digital values between 0 and 1024 by the ADCs, they were simply scaled between 0 and 1 to hasten the training speed. For the training phase, the normalized readings are used as training patterns to generate the user specific model. Any readings obtained during the testing phase are similarly processed and fed through the model to generate an instantaneous real-time classification. The overall processing scheme and the flow of model information between the training and testing phases of system operation are shown in Figure 3.

III. RESULTS

A. Training Data

A summary of the data collected during the training phase, including training speeds (discounting calibration time) and training patterns, is shown in Table 3. Pattern 1 corresponds to the training patterns for Finger Extension, Patterns 2-4 the samples for Finger Flexion, Patterns 5-7 for Wrist Flexion, and Patterns 8-10 for Wrist Extension. Note that for viewing convenience, the sensor values are shown as scaled integer values between 0 and 100.

TABLE III. TRAINING DATA

Pattern	Participant 1			Participant 2			Participant 3		
	Channels								
	1	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
1	0	0	0	0	0	0	0	0	0
2	99	99	34	99	99	8	99	0	99
3	99	99	40	99	99	14	99	0	93
4	99	99	34	99	99	6	99	0	99
5	91	46	19	99	99	88	98	99	42
6	85	43	19	99	99	83	98	99	49
7	82	39	16	99	99	77	99	99	54
8	12	5	0	31	0	0	0	0	22
9	5	4	0	26	0	0	0	58	22
10	10	5	0	34	0	0	0	54	17
Training Speed (s)	181.63			48.51			52.95		
Average Training Speed (s)	94.36								

B. Real-Time Classification Accuracy Analysis

During the testing phase, the real-time model predictions were recorded along with the given action commands for offline analysis. Plots of the results of the classification experiments for the subject are shown in Figure 4. The class labels found on the y-axis follow the numbering scheme presented in Table 1. The red graph tracks the action commands given to the user, and hence the "expected" classification, and the blue graph tracks the actual real-time prediction given by the model. From visual inspection it can be seen that the two curves were matched in general, with only a slight time delay between the two. This delay was attributed to the reaction time the user needed to process the instruction given on the monitor and the time needed to move their hand into the following hand configuration. In general, the delay took somewhere between 0.3 and 0.9 seconds.

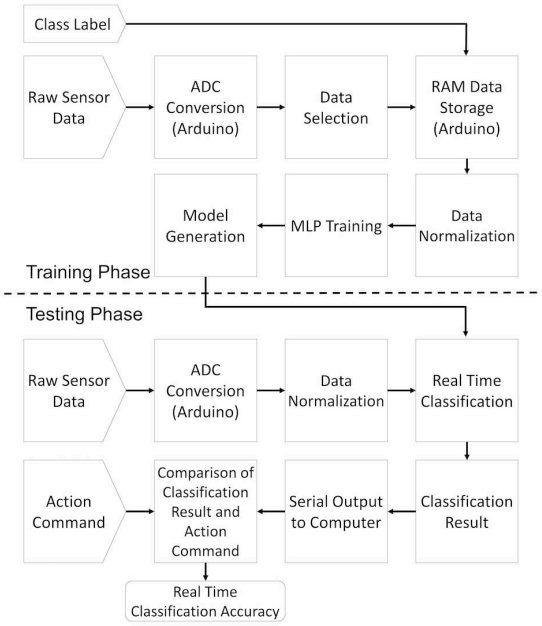

Fig. 3. Data Processing Sequence.

Note that "Section 1 Delay" refers to the time delay in the transition between configurations 4 and 1. For each section, the time delay, t is calculated by counting the number of samples, n since the new instruction was given that still retained the previous classification. In other words, in each 10-second epoch, the time delay was considered to be the number of samples classified as the previous action command. The sampling rate in this 10-second interval was then used to calculate the time delay, and the delays for each configuration for all three sections were averaged to obtain the final value displayed in the Table 4.

TABLE IV. TIME DELAY FOR ACTION RESPONSE

Delay	Section 1		Section 2		Section 3		Section 4	
Subject	t (s)	n	t (s)	n	t (s)	n	t (s)	n
1	0.35	5	0.38	5	0.74	9	0.54	7
2	0.64	9	0.83	12	0.72	9	0.70	9
3	0.35	5	0.47	7	0.54	7	0.51	6
Average	0.45	6.33	0.56	8	0.67	8.33	0.58	7.33
Stdev	0.17	2.31	0.24	3.61	0.11	1.15	0.10	1.53

In order to properly account for this time delay, the classification accuracy was analyzed by first removing a number of samples from each 10 second interval equal to the average time delay for that second. However, this data analysis scheme gives rise to a source of consistent error in the transition state classification accuracy. The program is implemented in such a way that the displayed classification only changes once the model predicts a new class twice (consecutively), a protocol we call "double confirmation," which is aimed at eliminating some of the random variations in muscle activations reported in [9]. However, this leads to some overestimation of classification accuracy in the transitions that are relatively quick. On the other hand, simply counting the number of samples until a "new" classification also yields potential underestimation because there is some transition time needed for the user to move

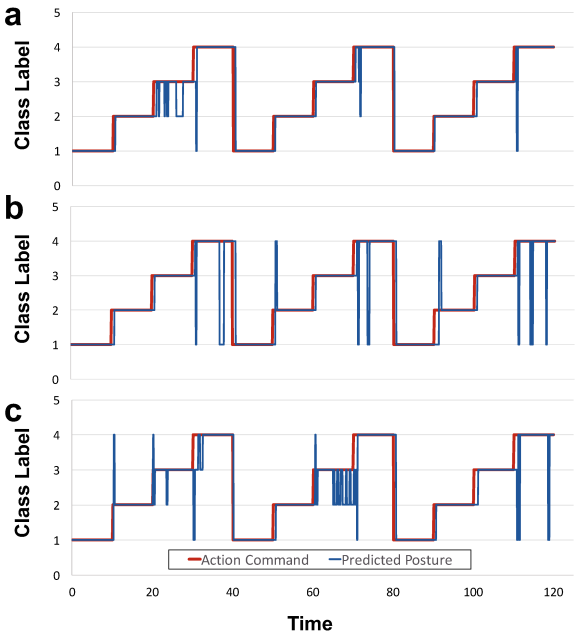

Fig. 4. Real-Time Classification Results for a) Subject 1; b) Subject 2; c) Subject3.

between different configurations. This effect is most pronounced in the transition between wrist flexion and wrist extension, because the user needs to "pass through" the finger extension configuration in between.

The results of the classification accuracy analysis are presented in Table 5. The overall accuracy across all classes was 94.94%. Across all four classes, finger extension was unsurprisingly classified the most accurately at 99.52%. Finger flexion was also classified very strongly at 98.38%. Both classes 3 and 4, however, were classified with much more variable performance. Wrist flexion was classified at 91.04%. Wrist extension was classified at 90.83%.

TABLE V. REAL TIME CLASSIFICATION ACCURACY

	Accuracy (%)				
Subject	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Overall**
1	99.52	99.26	90.34	96.26	96.35
2	99.76	96.63	100	87.25	95.91
3	99.29	99.26	82.77	88.98	92.58
Average	99.52	98.38	91.04	90.83	94.94
Stdev	0.24	1.52	8.64	4.78	2.06

The difficulty with classifying classes 3 and 4 are probably due to similarities in readings between different classes. By looking at the training data in Table 3, it is observed that the training patterns between classes 2 and 3 are relatively similar, as are the training patterns for classes 1 and 4. This similarity highlights the importance of maintaining the same arm posture, including arm angle, for the duration of the trial, as well as maintaining good sensor contact with the FMG band in order to replicate results.

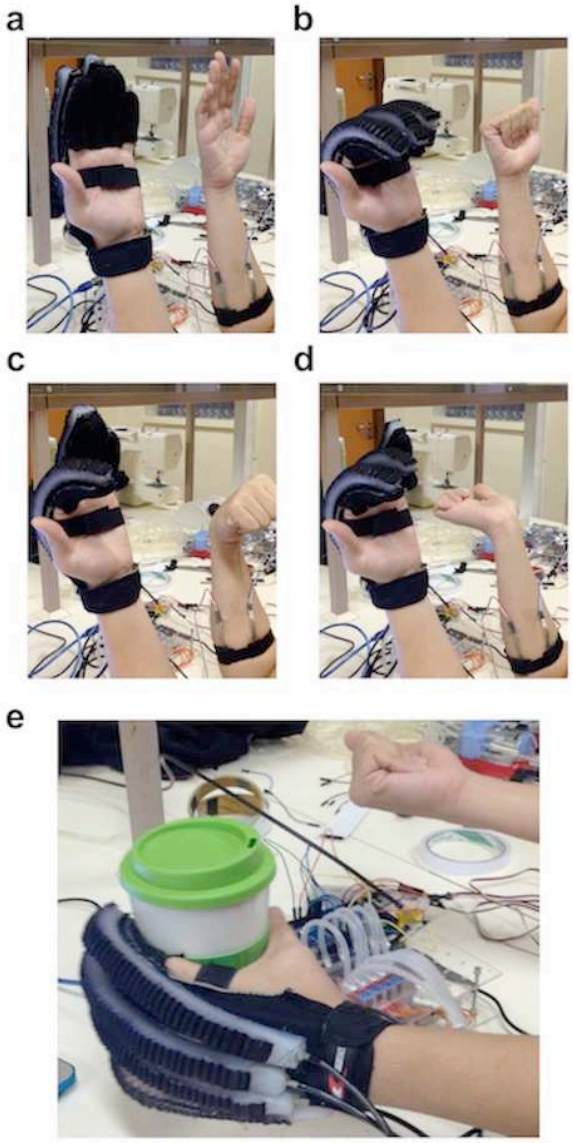

Fig. 5. Hand Configurations and Mapped Glove Actions. a) Class 1, b) Class 2, c) Class 3, and d) Class 4. e) The subject transferred a tumbler with a soft robotic glove and the proposed FMG system.

IV. USABILITY EXPERIMENT

A final experiment was devised to investigate the ability of the control system to provide the user with intuitive control of a soft robotic glove [17]. One healthy subject (Subject 1), who also participated in the classification experiment, was asked to wear the FMG band on the right arm and the glove on the left in a paired hand movement strategy (Fig. 5). The subject was asked to first wear the FMG band to detect class 2 (four finger flexion) and use the glove to transfer a tumbler back and forth 10 times between two targets 40 cm apart, making sure to grip and release each time (Fig. 5e). The same subject was then asked to perform the same task manually using pushbuttons to grip and release. Both trials were timed for comparison.

The subject was able to successfully transfer the tumbler between the two targets in 110.66 seconds. When the same subject was asked to repeat the same task using the

pushbuttons to activate the grasping action instead of the band, the task was performed only slightly faster, in 108.58 seconds. Using the latter trial as a baseline, it can be seen that using the glove to classify the user intention resulted only in an approximately 1.92% increase in operation time.

While it seems that the classification performance of the system is good in general, there are several limitations to the current setup. On the side of the experimental setup, we note that the classification accuracy analysis scheme was designed to handle the need to deal with non-negligible user reaction and transition time. Future experimental setups may utilize an LED-based instruction system as opposed to a Serial output instruction system to improve the user's reaction speed and sampling rate.

Additionally, we note that the current data is obtained only from healthy subjects, and that the usability experiment was conducted with the FMG band on one hand and the glove on the other. While the results are promising for the FMG approach to intention-driven hand rehabilitation, it is at this time unclear whether the system would be as successful on real rehabilitation patients without testing. Hence, data should be collected from actual patients. The data in this study is also limited in the sense that the user, especially for the wrist flexion and wrist extension configurations, needed to retain a consistent motion and posture in order to retain classification accuracy. If the user trained the model by laying their arm on the table at a 60-degree angle, for example, the results would have been much worse if the user angled their arm to 90 degrees during the testing phase. This limitation represents something of an impediment to a higher level of user intuitive system operation, but this issue may be mitigated by adjusting the training phase so that the readings for each configuration in several different arm postures are all sampled as training data. As a side note, the creation of an integrated electronics piece and eliminating the need for wires might also improve the user's experience and ultimately the system performance.

V. Conclusion

Research was performed towards the development of a simple and straightforward wearable device for classifying force myographic (FMG) signals for user intention-based control during hand rehabilitation with a soft robotic glove. An adjustable, tight form-fitting Velcro strap with three force sensitive resistor (FSR) sensors was developed to capture signals from the three major forearm muscle groups. We used the multilayer perceptron (MLP) to classify the raw 3-channel sensor data into four different hand configurations. Three healthy volunteer performed a testing protocol to evaluate the classification accuracy of the system. The average real-time testing accuracy across all four classes was 94.04% with a standard deviation of 2.06%. A usability experiment in which one participant wore the strap to control a soft robotic glove with the other hand was also conducted. The completion time for an item transfer task using the system was only 1.92% longer than by manually selecting hand configurations using buttons. These results indicate the feasibility of FMG and the proposed classification system as a control mechanism and tool for hand rehabilitation.

Acknowledgement

The authors would like to thank NUS Graduate School for Integrative Sciences and Engineering for providing scholarship support to the first author of this paper.

References

[1] C. E. Lang, M. D. Bland, R. R. Bailey, S. Y. Schaefer, and R. L. Birkenmeier, "Assessment of upper extremity impairment, function, and activity after stroke: foundations for clinical decision making," *J Hand Ther,* vol. 26, pp. 104-14, 2013.

[2] L. Rosenstein, A. L. Ridgel, A. Thota, B. Samame, and J. L. Alberts, "Effects of combined robotic therapy and repetitive-task practice on upper-extremity function in a patient with chronic stroke," *Am J Occup Ther,* vol. 62, pp. 28-35, 2008.

[3] P. Maciejasz, J. Eschweiler, K. Gerlach-Hahn, A. Jansen-Troy, and S. Leonhardt, "A survey on robotic devices for upper limb rehabilitation," *Journal of NeuroEngineering and Rehabilitation,* vol. 11, p. 3, 2014.

[4] P. Polygerinos, Z. Wang, K. C. Galloway, R. J. Wood, and C. J. Walsh, "Soft robotic glove for combined assistance and at-home rehabilitation," *Robotics and Autonomous Systems,* vol. 73, pp. 135-143, Nov 2015.

[5] P. Polygerinos, K. C. Galloway, E. Savage, M. Herman, K. O. Donnell, and C. J. Walsh, "Soft robotic glove for hand rehabilitation and task specific training," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2913-2919.

[6] H. K. Yap, J. C. H. Goh, and C. H. Yeow, "Design and Characterization of Soft Actuator for Hand Rehabilitation Application," in *6th European Conference of the International Federation for Medical and Biological Engineering*, 2015, pp. 367-370.

[7] H. K. Yap, J. H. Lim, F. Nasrallah, J. C. H. Goh, and R. C. H. Yeow, "A soft exoskeleton for hand assistive and rehabilitation application using pneumatic actuators with variable stiffness," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4967-4972.

[8] H. K. Yap, B. W. K. Ang, J. H. Lim, J. C. H. Goh, and R. C. H. Yeow, "A Fabric-Regulated Soft Robotic Glove with User Intent Detection using EMG and RFID for Hand Assistive Application," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.

[9] H. K. Yap, J. H. Lim, F. Nasrallah, J. Cho Hong Goh, and C. H. Yeow, "Characterisation and evaluation of soft elastomeric actuators for hand assistive and rehabilitation applications," *J Med Eng Technol,* vol. 40, pp. 199-209, May 2016.

[10] J. C. Yeo, H. K. Yap, W. Xi, Z. Wang, C.-H. Yeow, and C. T. Lim, "Flexible and Stretchable Strain Sensing Actuator for Wearable Soft Robotic Applications," *Advanced Materials Technologies,* 2016.

[11] L. Dipietro, A. M. Sabatini, and P. Dario, "Artificial neural network model of the mapping between electromyographic activation and trajectory patterns in free-arm movements," *Med Biol Eng Comput,* vol. 41, pp. 124-32, Mar 2003.

[12] O. Amft, G. Troster, P. Lukowicz, and C. Schuster, "Sensing Muscle Activities with Body-Worn Sensors," presented at the Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks, 2006.

[13] Z. G. Xiao and C. Menon, "Towards the development of a wearable feedback system for monitoring the activities of the upper-extremities," *J Neuroeng Rehabil,* vol. 11, p. 2, 2014.

[14] C.-M. Kuan, "Artificial Neural Networks."

[15] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks,* vol. 2, pp. 359-366, 1989.

[16] D. E. Rumelhart, J. L. McClelland, and S. D. P. R. G. University of California, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*: MIT Press, 1986.

[17] H. K. Yap, J. H. Lim, F. Nasrallah, F.-Z. Low, J. C. H. Goh, and R. C.-H. Yeow, "MRC-Glove: A fMRI Compatible Soft Robotic Glove for Hand Rehabilitation Application," in *IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2015, pp. 735-740.