Hand Gesture Recognition Systems with the Wearable Myo Armband

Engin Kaya, Tufan Kumbasar
Control and Automation Engineering Department
Istanbul Technical University
Istanbul, Turkey
{kayaen, kumbasart}@itu.edu.tr

***Abstract*—The hand gesture recognition systems deal with identifying a given gesture performed by the hand. This work addresses a hand gesture recognition method to classify and recognize the numbers from 0 to 9 in Turkish Sign Language based on surface electromyography (EMG) signals collected from a wearable device, namely the Myo armband. To accomplish such a goal, we have utilized machine learning techniques to recognize the hand gestures. In this context, seven different time domain features are extracted from the raw EMG signals using sliding window approach to get distinctive information. Then, the dimension of the feature matrix is reduced by using the principal component analysis to reduce the complexity of the deployed machine learning methods. The presented study includes the design, deployment and comparison of the machine learning algorithms that are k-nearest neighbor, support vector machines and artificial neural network. The results of the comparative comparison show that the support vector machines classifier based system results with the highest recognition rate.**

Keywords—hand gesture recognition; k-nearest neighbor; support vector machines; artificial neural networks; Myo armband

I. INTRODUCTION

Humans interact with technological devices using mouse, keyboard, joystick or touchpad. That input methods restrict the people, for that reason Human Computer Interaction researches focus on creating more natural interfaces between human and computer [1]. In that point, the hand gestures arise as a new interface for devices. There are various methods for hand gesture detection such as cameras [1], gloves [2] and electrodes [3]. Although visual based solutions provide high recognition rate, they face difficulties such as light conditions, skin and background color, spatial position and orientation of the fingers [1]. On the other hand, glove systems hinder people from using their hands comfortably [4]. Thus, electrode based systems are appeared as an alternative solution. When muscle cells contract to perform a hand gesture, they generate an electrical signal that is the myoelectric signal [5]. The technique of recording and study of myoelectric signal is defined as electromyography (EMG). The electrodes, located on forearm, can measure directly the EMG signals and thus the EMG apparatus are easy to use and are not effected by light, color and sight view as the visual based solutions for hand gesture recognition systems.

In literature, there are two main types of electrodes, which are needle electrodes and surface electrodes [6]. From an application-oriented view, the surface electrodes facilitate the implementation and are more convenient [7]-[13]. In these studies, it has been shown that it is not possible to construct a basic EMG based hand gesture recognition by using a threshold, since the problem is quite complex due to various gestures and also the EMG systems have multiple electrodes. Thus, to end up with an efficient and generic solution, machine learning methods have been used to infer the underlying pattern of the signals recognize automatically to hand gestures. To accomplish such a goal, k-Nearest Neighbor (kNN) [8], Decision Trees [9], Support Vector Machine (SVM) [10], [11] and Artificial Neural Networks (ANN) [12], [13] are employed as they have shown to be highly efficient.

In this study, we will present machine learning based hand gesture recognition systems with the wearable Myo armband to classify the numbers from 0 to 9 in Turkish Sign Language (TSL). We will first briefly introduce the Myo armband that has 8 electrodes/channel and handled data set. Then, we will utilize machine learning techniques to recognize the hand gestures. In this context, a sliding window approach will be applied for signals of each channel of the Myo armband and seven time domain features will be extracted from each obtained window. Hence, the feature matrix will have 56 dimensions that will be then reduced by applying Principal Component Analysis (PCA) algorithm to 15 dimensions. Then, using the processed feature set, we will design and deploy the machine learning algorithms kNN, SVM and ANN as classifiers and compare their recognition performances. In the comparative comparisons, the performance of the kNN algorithm will be examined for various distance metrics and number of neighbors while the SVM classifier will be tested for linear and radial basis function kernel methods. For the SVM, the performances of One Versus One (OVO) and One Versus All (OVA) binary classification methods will also be examined. We will examine the performance of the ANN for various numbers of neurons in hidden layer and three different training algorithm. In the light of these analyses, we will adjust the tuning parameters of the classification methods to optimal values to perform overall performance comparison in hand gesture recognition. The results will show that highest recognition rate will be obtained when the SVM classifier is deployed.

This paper is organized as follows. Section II introduces the Myo armband and handled data set. Section III describes the proposed hand gesture recognition system. The comparative performance results of the proposed systems are given in Section IV. Chapter V present the conclusions and future work.

978-1-5386-7641-7/18/$31.00 ©2018 IEEE

II. Myo Armband and Hand Gestures

The Myo armband, launched by Thalmic Labs on 2013, is a wearable and Bluetooth based wireless device that contains ARM Cortex M4 Processor, eight EMG sensor and nine axis Internal Measurement Unit including gyroscope, accelerometer and magnetometer [14]. The device provides feedback to user via LED lights and vibrations. As shown in Fig. 1, users wear the Myo armband on the forearm. The EMG data is measured from 8 channel of electrodes, whose locations are seen in Fig. 1. The measured raw EMG data related with muscle activity can be collected via the SDK of device with sampling rate of 200 Hz [15]. In this work, we have used the Matlab Mex Wrapper [16] that records the EMG signals in a normalized range of [-1 1].

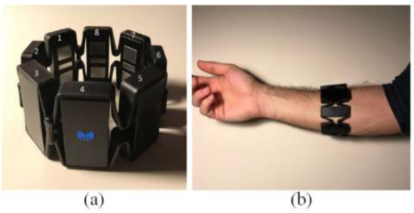

Fig. 1. The Myo armband (a) channel placement, (b) worn on forearm

We have collected a hand gesture data set consisting of 10 gestures that are numbers between 0 and 9 in the TSL and one gesture defined as a resting position. Fig. 2 illustrates the gestures of the handled numbers in TSL [17]. A user performed and hold on the gesture during the collection of the signal; therefore, the transition between gestures were not considered. We have performed 5 experiments and collected 5x1000 data points from each channel of the Myo armband, thus a total of 11x5000 data points were collected for training the machine learning methods.

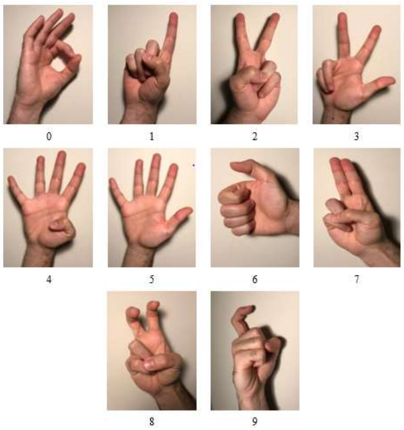

Fig. 2. Handled gesture set, numbers between 0 and 9 in TSL

III. Hand Gesture Recognition Systems

In this section, we will present the developed hand gesture recognition system based on the machine learning methods, namely kNN, SVM and ANN. The design of the system follows the steps of machine learning application. First, the features are extracted from recorded signal to get distinctive information. Then, the number of dimension in feature matrix is reduced to facilitate the classification process. Then, the classification algorithms are trained to recognize the hand gesture. The pipeline of the algorithm is illustrated in Fig. 3.

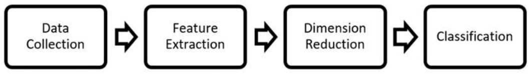

Fig. 3. Classification algorithm pipeline

A. Feature Extraction

The first step of the process is to extract meaningful and distinctive features from the raw data. Due to the complex and noisy characteristics of EMG, proper selection of features is essential for classification. In machine learning, working with directly raw data increases the complexity of implementation and processing cost. Also, some hidden patterns which cannot be visible in a single data point can be obtained in features. In this study, we have performed a sliding window method to record the data with windows length of 40 and sliding length of 20 as a preprocessing step. Then, the seven time domain features are calculated which are the Mean Absolute Value (MAV), variance, waveform length, Root Mean Square (RMS), Willison amplitude, Zero Crossing (ZC) and Slope Sign Change (SSC) of the signals.

1) Mean Absolute Value

The MAV value presents the contraction level of muscles and is calculated as follows:

$$Y = \frac{1}{N}\sum_{i=1}^{N}|x_i| \tag{1}$$

2) Variance

The variance of a signal is defined as,

$$Y = \frac{1}{N-1}\sum_{i=1}^{N}(x_i{}^2) \tag{2}$$

3) Waveform Length

The waveform length is a measure of the cumulative variations and is expressed as,

$$Y = \sum_{i=1}^{N-1}(|x_{i+1} - x_i|) \tag{3}$$

4) Root Mean Square

RMS is another well-known feature and defined as,

$$Y = \frac{1}{N}\sqrt{\sum_{i=1}^{N} x_i{}^2} \tag{4}$$

5) Willison Amplitude

This feature counts how many times the contiguous time instances in signal exceeds a threshold (thr) and is defined as:

$$Y = \sum_{i=1}^{N-1} f(|x_{i+1} - x_i|) \tag{5}$$

where

$$f(x) = \begin{cases} 1, & if\ x > thr \\ 0, & otherwise \end{cases} \tag{6}$$

The *thr* value is experimentally found as 0.2.

6) Zero Crossing

ZC counts that how many times the signal crosses zero and is defined as,

$$Y = \sum_{i=1}^{N-1} f(x_i, x_{i+1}) \tag{7}$$

where

$$f(x_i, x_{i+1}) = \begin{cases} 1, & if\ (x_i > thr\ \&\ x_{i+1} < thr) \\ & \quad or\ (x_i < thr\ \&\ x_{i+1} > thr) \\ 0, & otherwise \end{cases} \tag{8}$$

The *thr* value for ZC is obtained experimentally and set as 0.3.

7) Slope Sign Change

SSC is a measure of changing the sign of the slope of the signal. The mathematical expression of SSC is,

$$Y = \sum_{i=2}^{N-1} f(x_{i-1}, x_i, x_{i+1}) \tag{9}$$

where

$$f(x_{i-1}, x_i, x_{i+1}) = \begin{cases} 1, & if\ (x_i < x_{i+1}\ \&\ x_i < x_{i-1}) \\ & \quad or\ (x_i > x_{i+1}\ \&\ x_i > x_{i-1}) \\ 0, & otherwise \end{cases} \tag{10}$$

Remark: All seven features were extracted from each channel of the recorded data; therefore, at the end of the process, 56 dimensional feature matrix were obtained.

B. Dimension Reduction

Dimension reduction is a process of reducing the number of variables in the feature vector set. In the classification problems, choosing the number of inputs has great importance to determine the time and space complexity of the classifier; therefore, working with less dimensional data provides simpler solution [18]. In this study, the well-known technique PCA [19], has been employed to reduce the dimension of the feature matrix. We have observed that the first 15 principle components preserve about 96 percent of the variance; and thus we have reduced the feature matrix dimension from 56 to 15.

C. Classfication Algorithms

In this work, as we have 11 gestures to be recognized, the classification problem has to be defined as a multiclass one. In this context, we have designed and deployed kNN, SVM and ANN algorithms as classifiers. The kNN and SVM were deployed using Matlab Statistics and Machine Learning Toolbox [20] while the ANN with Matlab Neural Network Toolbox [21]. The remainder of this section gives background of algorithms and details how the algorithms are designed.

1) k-Nearest Neighbour Algorithm

kNN algorithm is a non-parametric instance-based lazy learning algorithm [22]. Each labelled instance in the training set is defined as $\{x_i, y_i\}$, where x_i represents the attribute-value and y_i is class label [18]. When the algorithm takes a test point x, the output label y^* can be obtained via the algorithm given in Table 1. Here, the distance $D(x, x_i)$ to be calculated can be the Euclidean, City Block, Minkowski and Chebyshev distance metrics [23] and are defined in Table 2.

Table 1. kNN Algorithm

1. Calculate distance $\boldsymbol{D(x, x_i)}$ for each training instance
2. Sort the distances
3. Select k closest neighbors with labels
4. Assign majority voted label as output, $\boldsymbol{y}^*$

In this section, we will investigate the effect the parameter k and the distances $D(x, x_i)$ given in Table 2 on the classification performance. Thus, increasing k numbers from 1 to 10 for four distance metrics were tested and the results are shown in Fig. 4. It can be seen that increasing the value of k improves the accuracy for all the distance metrics. The distance metrics of Euclidean and Minkowski give almost same accuracy, but Chebychev distance resulted with the lowest one. Cityblock distance metric has as better performance than others; hence, it is selected as best distance metric with optimal number of neighbor as k =6.

Table 2. Distance Metrics

Euclidean Distance:	$\boldsymbol{D(x, x_i) = \sqrt{\sum_{i=1}^{k} (x - x_i)^2}}$
City Block Distance:	$\boldsymbol{D(x, x_i) = \sum_{i=1}^{k} \lvert x - x_i \rvert}$
Minkowski Distance:	$\boldsymbol{D(x, x_i) = \sqrt[3]{\sum_{i=1}^{n} \lvert x - x_i \rvert^3}}$
Chebyshev Distance:	$\boldsymbol{D(x, x_i) = \max_i\{\lvert x - x_i \rvert\}}$

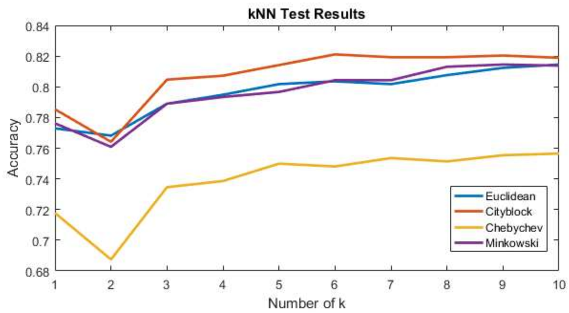

Fig. 4. Accuracy of kNN for different k values and $D(x, x_i)$

2) Support Vector Machine Algorithm

One of the most popular learning algorithm is the SVM which basically separates the data by finding an optimal hyperplane. The optimal hyperplane is constructed using the support vectors which are the closest points in the two classes each other. The gap between the hyperplane and support vectors is called margin (w) and SVM aims to find the maximum margin for better separation. The optimization problem of the SVM is defined as:

$$\min_{w} \frac{1}{2} \|w\|^2 \tag{11}$$

subject to

$$y_i(w^T \phi(x_i) + b) - 1 \geq 0\ , \forall_i \tag{12}$$

where $y \epsilon\{-1,1\}$, x is input, b is bias and $\phi()$ maps x into a higher dimensional space. Once the optimization is completed, the optimal w satisfies:

$$w = \sum_{i=1}^{l} y_i \alpha_i \, \phi(x_i) \tag{13}$$

where l is the number of support vectors and α is Lagrange multiplayer. Now, when a query instance x_q arrives, the SVM decides the output by:

$$f(x_q) = sign\left(\sum_{i=1}^{l} y_i \alpha_i \, K(x_q, x_i) + b\right) \tag{14}$$

where $K()$ is the kernel function which ensures non-linear property to SVM by transforming the input data to higher dimensional space. If $K()$ is chosen as in (15), then it is called linear SVM.

$$K(x_i, x_j) = x_i \cdot x_j \tag{15}$$

In addition, a Radial Basis Function (RBF) can be also employed.

$$K(x_i, x_j) = \exp(-\frac{|x_i - x_j|^2}{2\sigma^2}) \tag{16}$$

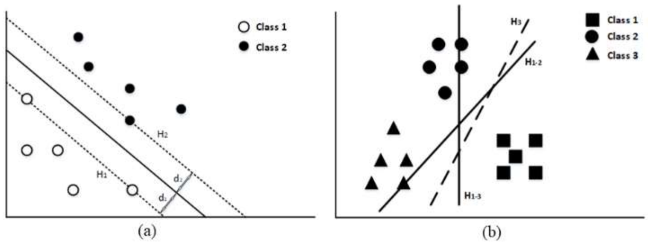

Fig. 5. (a) Support vectors and margin (b) OVO and OVA methods

We have firstly examined the accuracy of the linear and RBF based SVM methods by repeating the experiments 10 times. As it can be seen from Fig. 6, the linear SVM provided a better recognition rate for the handled data set.

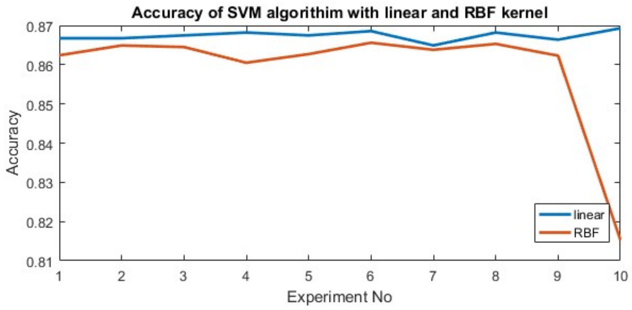

Fig. 6. Comparison of the linear and RBF kernels

As it was mentioned, SVM is binary classifier; however, the dataset in our case, is multiclass. Hence, the methods, OVO and OVA can be used to obtain multiclass outputs from binary classifiers. OVO trains a separate classifier for each the class pairs in the training set. On the other hand, OVA trains one classifier for each class by considering pairs as the selected class and rest of all. The OVO and OVA methods are illustrated in Fig. 5b that includes example dataset with three classes for linear SVM. The performances of OVO and OVA methods for the data set is given in Fig. 7. It can be observed that the linear SVM employing the OVO resulted with a higher accuracy when compared to its OVA counterpart.

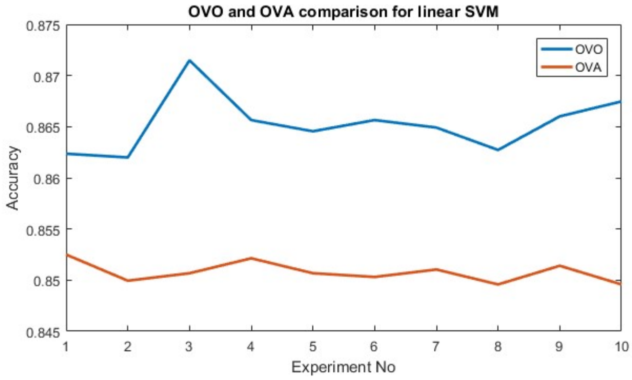

Fig. 7. Comparison of the OVO and OVA methods

3) *Artificial Neural Networks*

Most of the applications of ANNs use the multi layered perceptron (MLP) model [24]. As shown in Fig. 8, each layer is fully connected to next layer; therefore, the output value of a neuron in each layer is an input value of the neurons in the next layer [25]. Here, a single neuron is defined as:

$$a = g\left(\sum_{i=0}^{N} u_i w_i\right) \tag{17}$$

where u, w and a refer input, weight and output of the neuron, respectively. $g()$ denotes the activation function which can be linear and nonlinear such as *sigmoid*, *hyperbolic tangent sigmoid* and *softmax*. The training of the multilayer ANN is usually accomplished with backpropagation algorithms to minimize the cross entropy performance value.

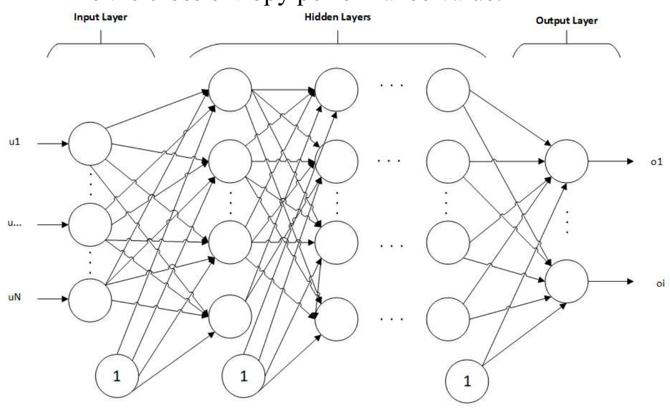

Fig. 8. Structure of MLP

In this study, the MLP ANN is configured with one hidden layer defined with *tansig* activation function while the output layer is defined with the *softmax* activation function. The performance of the MLP has been investigated for increasing number of neurons from 5 to 55 in hidden layer and for the training functions; gradient descent backpropagation (*traingd*), Scaled Conjugate Gradient (*traingscg*) and Resilient Backpropagation (*trainrp*). The resulting accuracies of these structures are given in Fig. 9. It can be firstly observed that increasing the number of neurons in the hidden layer has not significantly improved the recognition accuracy. For the handled data set, an optimal number of hidden of neurons can be set as 12. In comparison of the training methods, the *traingd* method resulted with lowest accuracy; whereas, *trainrp* and *trainscg* provided significantly better performances. The *trainrp* function has been chosen as optimal training function as it has resulted with slightly better performance in comparison to the *trainscg* one.

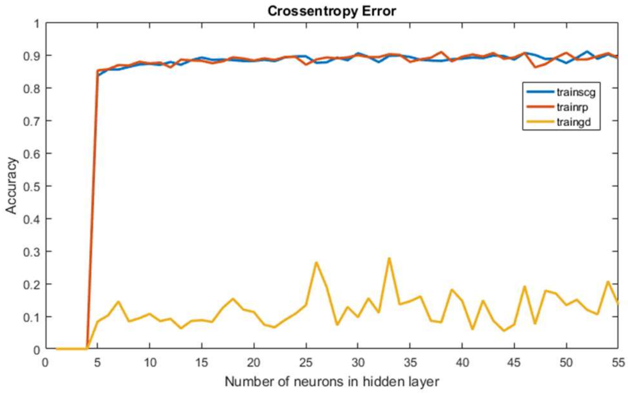

Fig. 9. Accuracies of ANN methods for various configurations

IV. Comparative Experimental Results

In this section, the performance of kNN, SVM and ANN will be compared. First, the parameters of classification algorithms are set to optimal values that are found in Section III.C. Then, comparison of the algorithms will be accomplished in terms of overall performance and each classes performance.

In the performance measures of multi-class classification, first, the confusion matrix is obtained using the known target values and the outputs values of the classifier. Then, evaluation metrics, which are precision, recall and Fscore, are computed via the confusion matrix [26]. The precision is the ratio between number of correctly classified positive instances and outputs of the classifier which are labelled as positive. The recall is the ratio between number of correctly classified positive instances and number of the positive classes in target. The weighted average of the precision and the recall is interpreted as Fscore which reaches its best value at 1 and worst value at 0. The mathematical expressions of precision, recall and Fscore metrics are given in Table 3. Here, tp_i, fp_i and fn_i denotes the true positive, false positive and false negative counts of class i, respectively. The coefficient β determines the balance between P and R to calculate F, and was chosen as 1 in this work.

Table 3. Definition of Evaluation Metrics

Measure	**Formula**
Precision	$P = \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l}(tp_i + fp_i)}$
Recall	$R = \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l}(tp_i + fn_i)}$
Fscore	$F = \frac{R(\beta^2 + 1)P}{\beta^2 P + R}$

The validation phase of the training has been performed by employing the k-fold in each experiment. According that, the input dataset is divided into t subdata set, where each subdata set has equal number of instance and labelled by randomly from 1 to t. The classification algorithms are run for t times and each time one of the subdata is chosen for test and the rest is used for training. At the end of t times, the average of each calculations of metrics are considered as output. Number of fold was selected 10 for this work. Table 4 includes the Fscore ratios, which were obtained by averaging the 10 experiments.

Table 4. Fscores of classifiers

Classifier	kNN	SVM	ANN
Fscore	0.8167	0.8661	0.8520

The results in Table 4 shows that, all the classifiers have more than a 0.8 Fscore ratio; whereas, the kNN algorithm has lowest recognition rate. Although the ANN based method improved the performance, the highest Fscore ratio is obtained for SVM algorithm. In addition to overall performance, the recognition rate of each class is also investigated in terms of recall, precision and Fscore. According to results given in Table 5, for all classifiers, the classes 'rest' and class '2' have highest and lowest recall ratio, respectively. The highest precision ratio of all classifiers is achieved for the class '9'; whereas, SVM and ANN resulted also with equal ratios with class '9' for classes '6', '7' and '8'. Fscore ratio of class '6', '7', '8' and '9' were obtained as higher than the other classes for all algorithms. As a result, by looking the Fscore ratios, the linear SVM classifier provided highest recognition rate for all classes.

Table 5. Performance of each hand gesture class

		Hand Gesture Classes										
		Rest	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Recall	**kNN**	0.97	0.93	0.84	0.80	0.80	0.82	0.83	0.84	0.82	0.82	0.82
	SVM	0.99	0.92	0.88	0.86	0.87	0.87	0.87	0.88	0.87	0.87	0.87
	ANN	0.95	0.91	0.85	0.82	0.84	0.85	0.85	0.86	0.85	0.85	0.85
Precisio	**kNN**	0.64	0.70	0.68	0.67	0.70	0.73	0.75	0.77	0.77	0.78	0.79
	SVM	0.71	0.77	0.78	0.78	0.81	0.82	0.84	0.85	0.85	0.85	0.85
	ANN	0.72	0.77	0.77	0.76	0.79	0.80	0.82	0.83	0.83	0.83	0.83
Fscore	**kNN**	0.77	0.80	0.75	0.73	0.75	0.77	0.79	0.80	0.80	0.80	0.80
	SVM	0.83	0.84	0.83	0.82	0.83	0.84	0.85	0.86	0.86	0.86	0.86
	ANN	0.82	0.83	0.80	0.78	0.81	0.81	0.83	0.84	0.84	0.84	0.84

V. Conclusions

In this paper, we have presented hand gesture recognition systems that deals with identifying a given gesture performed by the hand with the aids of machine learning methods. This work addressed a hand gesture recognition method to classify and recognize the numbers from 0 to 9 in TSL based EMG signals collected from the Myo armband. To recognize the hand gestures, we have utilized machine learning techniques kNN, SVM and ANN. In this context, seven time domain features were extracted from each channel of the Myo armband; and thus a feature matrix was obtained with 56 dimensions. PCA algorithm was applied to reduce the dimension. Then, obtained feature matrix was classified using the kNN, SVM and ANN algorithms. As result of comparison, the SVM classifier was determined as best algorithm with 0.87 Fscore ratio.

Future work includes testing the proposed method with recording signals from different people and for more complicated hand gestures.

References

[1] Khan, R. Z., & Ibraheem, N. A. "Comparative study of hand gesture recognition system," *Computer Science & Information Technology (CS & IT)*, vol. 2, no. 3, pp. 203-213, 2012.

[2] Bansal, B., "Gesture recognition: a survey," *International Journal of Computer Applications*, vol. 139, no. 2, pp. 8-10, 2016.

[3] Kim, Jonghwa, Stephan Mastnik, and Elisabeth André. "EMG-based hand gesture recognition for realtime biosignal interfacing," *Proceedings of the 13th international conference on Intelligent user interfaces*. ACM, 2008.

[4] Dipietro, L., Sabatini, A. M., & Dario, P., "A survey of glove-based systems and their applications," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 4, pp. 461-482, 2008.

[5] Bigland, B., & Lippold, O. C. J., "Motor unit activity in the voluntary contraction of human muscle," *The Journal of Physiology*, vol. 125, no. 2, pp. 322-335, 1954.

[6] Reaz, M. B. I., Hussain, M. S., & Mohd-Yasin, F., Techniques of EMG signal analysis: detection, processing, classification and applications. *Biological procedures online*, vol 8, no. 1, 2006.

[7] Catriona M. Steele, Janice W. Bennett, Sarah Chapman-Jay, Rebecca Cliffe Polacco, Sonja M. Molfenter and Mohamed Oshalla, *Applications of EMG in Clinical and Sports Medicine*, InTech, 2012.

[8] Benalcázar, M. E., Jaramillo, A. G., Zea, A., Páez, A., & Andaluz, V. H., "Hand gesture recognition using machine learning and the Myo Armband," *2017 25th European Signal Processing Conference (EUSIPCO),* IEEE, 2017.

[9] Ploengpit, Y., & Phienthrakul, T., "Rock-paper-scissors with Myo Armband pose detection," *2016 International Computer Science and Engineering Conference (ICSEC)*, pp. 1-5, December, 2016.

[10] Abreu, J. G., Teixeira, J. M., Figueiredo, L. S., & Teichrieb, V. "Evaluating Sign Language Recognition Using the Myo Armband", *2016 XVIII Symposium on Virtual and Augmented Reality (SVR)*, June, 2016.

[11] Krishnan, K. S., Saha, A., Ramachandran, S., & Kumar, S., "Recognition of human arm gestures using Myo Armband for the game of hand cricket," *2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS),* pp. 389-394, October, 2017.

[12] Atasoy, A., Kaya, E., Toptas, E., Kuchimov, S., Kaplanoglu, E., & Ozkan, M., "24 DOF EMG controlled hybrid actuated prosthetic hand", *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC),* August, 2016.

[13] Amatanon, V., Chanhang, S., Naiyanetr, P., & Thongpang, S., "Sign language-Thai alphabet conversion based on Electromyogram (EMG)," *The 7th Biomedical Engineering International Conference (BMEiCON),* pp. 1-4, IEEE, November, 2014.

[14] Thalmic Labs, Available: https://www.myo.com/techspecs, Date retrieved: 10.01.2018.

[15] Thalmic Labs, Available: https://developer.thalmic.com/, Date retrieved: 10.01.2018.

[16] MathWorks, *Myo SDK MATLAB MEX Wrapper*, Mark Tomaszewski, Available: https://www.mathworks.com/matlabcentral/fileexchange/55817-myo-sdk-matlab-mex-wrapper, Date retrieved: 10.01.2018.

[17] Güngör C., Bağrıaçık T., Demirdöğen İ., Günaydın A. and Karahan V. *Türk İşaret Dili Sözlüğü*, Ankara, 2015.

[18] Alpaydin, E., *Introduction to machine learning*. MIT press, 2014.

[19] Ghodsi, A., "Dimensionality reduction a short tutorial," Department of Statistics and Actuarial Science, Univ. of Waterloo, Ontario, Canada, 2006.

[20] MathWorks, *Statistics and Machine Learning Toolbox*, Available: https://www.mathworks.com/help/stats/index.html, Date retrieved: 10.04.2018.

[21] MathWorks, *Neural Network Toolbox*, Available: https://www.mathworks.com/help/nnet/index.html, Date retrieved: 10.04.2018.

[22] Cover, T. and Hart, P., "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21-27, 1967.

[23] Cha, S. H., "Comprehensive survey on distance/similarity measures between probability density functions," *City*, vol. 1, no. 2, pp. 1, 2007.

[24] Christopher M. B., *Neural networks for pattern recognition*. Oxford university press, 1995.

[25] LeCun, Y., Bottou, L., Orr, G. B., and Müller, K. R. "Efficient backprop," *Neural networks: Tricks of the trade*. pp. 9-50 Springer, Berlin, Heidelberg, 1998.

[26] Sokolova, M., & Lapalme, G., "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427-437, 2009.