A Real-time Gesture Prediction System Using Neural Networks and Multimodal Fusion based on data glove

Yunhao Ge and Bin Li,
Robotics Institute of Shanghai Jiao Tong University
Shanghai Jiao Tong University
Shanghai, China
gyhandy@sjtu.edu.cn, lbin_sjtu@sjtu.edu.cn

Weixin Yan and Yanzheng Zhao
State Key Lab of Mechanical System and Vibration
Shanghai Jiao Tong University
Shanghai, China

***Abstract*—Unlike static gesture recognition, a novel real-time gesture prediction system in this study can judge the intention of hand motion and predict the exact final gesture before the end of hand movement. Flex sensors are used to measure comprehensive motion data of data glove, which are positioned based on the biological muscle distribution characteristics of the hand. Position, velocity and acceleration information are extracted from raw data of data glove, while the adjacent finger-coupling features are also obtained by processing the position and velocity information. After data processing such as windowing and filtering, accuracy and effectiveness experiments are conducted to obtain the ideal features based on multimodal fusion. A combination of neural network and multiclass support vector machine (SVM) algorithms are used as prediction model. Neural network experiments are designed in which prediction time and accuracy are used as the optimization index to select the combination structure of the prediction model.**

Keywords—Gesture Prediction; Multimodal Fusion; Neural Network; Multiclass SVM; Flex sensors; Data glove

I. INTRODUCTION

As an effective and natural interface, human gesture recognition is widely used in interaction between humans and computational systems. Applications various from interaction with robotics [1,2] to sign language recognition [3], Mori et al. [4] pointed out that a real-time gesture recognition system had to be able to predict the gesture that is being executed before it ends. Gesture prediction has obtained an increasing amount of attention in pattern recognition and real-time systems. A few studies use the prediction concept gesture prediction, as described by Liu and Xiao [3]. In recent years, many studies were conducted by using digital camera [5] or depth sensors which provide three-dimensional depth data of the scene, such as Leap Motion controller [6] and Microsoft Kinect sensor [7,8]. However, some problems remain in the computer vision for gesture prediction, like significant computational and time costs for the algorithms [9]. Additionally, visual occlusion also affects the performance of gesture prediction based on vision system. Meanwhile, systems based on gloves or external sensors can effectively avoid the visual occlusion and high time costs.

978-1-5386-4362-4/18/$31.00 ©2018 IEEE

Preetham et al. [10] presented a gesture recognition glove prototype, while Jadhav, Joshi et al. [11] presented wearable sensing gloves along with flex sensors to determine four words in Indian sign language. But most of them focus on the static gesture recognition instead of gesture prediction. Our study combines the advantage of data glove and gesture prediction.

To make the prediction more effective and reliable, we proposed a novel embedded data glove real-time prediction system with multimodal fusion method and neural network prediction model, which replaces the traditional logical classification models such as if-else judgement or single traditional state estimation based on position [12]. Thus, the study in this paper of gesture prediction based on data glove is quite meaningful.

A significant amount of work on combining diverse feature types was applied to object and action recognition [13]. We use various data channels to describe each gesture at multiple scales not only spatially, but also temporally, during feature extraction to provide context for neural network gesture prediction.

In most practical applications, the late fusion of scores output by several models offers a cheap and surprisingly effective solution [14]. We use time cost and accuracy as optimization index to select the best combination of neural networks and obtain a prediction model.

II. SYSTEMATIC FRAMEWORK

The systematic framework of the proposed method (shown in Fig. 1) includes three steps: 1. Sensor location and data collection, 2. Multimodal fusion and feature extraction, and 3. Real-time prediction system.

Fig. 1. The systematic framework

A. Sensor Location and data collection

Position, precision and working condition of each flex sensor are specifically measured in data glove. First, studying

the biological characteristics of hand muscle helps to determine the exact direction of the moving joints of each finger and muscles. The flex sensors should coincide exactly with the flexor digitorum (shown in Fig. 2), especially the thumb. Adding the flex sensor in the vertical direction of the connection between aponeurosis palmaris and thenar can collect the palm movement. Hence, a total of six flex sensors an Arduino board and a wireless module are located on the data glove to collect the real-time raw movement of the hand. (shown in Fig. 3)

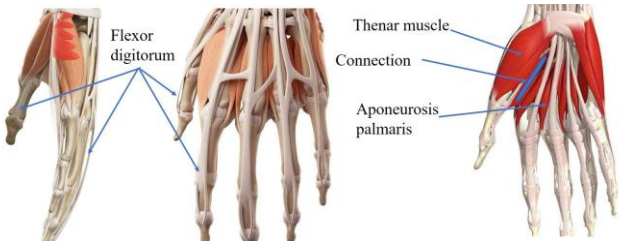

Fig. 2. The biological characteristics of the hand

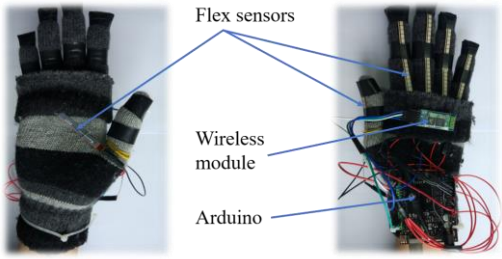

Fig. 3. The physical map of data-glove

We built hand gestures datasets, Flex-Gesture dataset, concluding sixteen common gestures: “scissor”, “rock”, “like”, “ok”, et, details are shown in Fig. 4. Each class of gesture contains 3000 six-dimension flex data for each sampling moment during the movement of hand. The flex data have corresponding labels, which can be used to training or testing machine learning algorithms.

Fig. 4. Illustrations of the sixteen hand gestures from the Flex-Gesture dataset. Left to right, top to bottom: one, three, four, six, seven, eight, little, German3, hold, like, ok, rock, scissor, paper, thumb, ring, yang et.

B. Feature Extraction and Multimodal Fusion

The data glove collects real-time position information of fingers movement and the bend of the palm, which conveys the spatial dimension. To extract more temporal information from the measured data, we calculate the relationship feature between the data as it various over time, the velocity and acceleration of each finger, and to make our system more stable, use the moving average method to smooth the velocity. Gaussian white noise is used for data augmentation, enlarging the dataset.

Velocity and acceleration of each finger calculation: To avoid the outliers and ensure the stability, we calculate the velocity of one moment with every six sampled data, while acceleration with four velocity data.

$$V_j^l = \frac{P_{j+m}^l - P_j^l}{m} \quad m = 6 \tag{1}$$

$$a_j^l = \frac{V_{j+n}^l - V_j^l}{n} \quad n = 4 \tag{2}$$

Where P is the position data collected from the flex sensors in data glove. The subscript j represents different moments, while l is the specific number of flex sensors.

Moving average method (equal to the low-pass filter): We select a window of a specific size, compute the average of all values in the window, and use the average as the center point of the window. This method is used to preprocess the velocity and acceleration of each finger, with a window of size 11, the context information is combined into the velocity V_i and acceleration a_i.

$$\hat{V}_i^l = \frac{1}{n+1} \sum_{j=i-n/2}^{i+n/2} V_j^l \tag{3}$$

$$\hat{a}_i^l = \frac{1}{n+1} \sum_{j=i-n/2}^{i+n/2} a_j^l \tag{4}$$

Where the width of the window is n+1.

Multimodal fusion with spatial, temporal and finger-coupling channels: Combine the flex sensor data and its velocity, such that every point of time corresponds to a multimodality vector. Use the adopt min-max method to scale the vector.

$$feature_i = \frac{feature_i - \min(feature_i)}{\max(feature_i) - \min(feature_i)} \tag{5}$$

To describe the interaction between adjacent fingertips, we propose double-finger features [15]. Eq.4 shows the absolute bend finger-distances between adjacent fingertips.

$$DisAf_i^l = \frac{\| P_i^l - P_i^{l+1} \|}{M}, \quad l = 1, \text{K}, \quad 4 \tag{6}$$

M is the range of finger positions. Note that, dividing by M normalizes the absolute bend fingertip-distance to the interval [0, 1]. Adjacent finger-distance features distinguish between the different types of interactions between adjacent fingertips. Position of each finger and palm reflect the spatial features, while velocity and acceleration reflect the temporal features, and finger adjacent coupling features are also calculated. The combination of these three different feature channels form a multimodal fusion.

With the method above, the six-dimension data in Flex-Gesture dataset is enlarged into n-dimension data, containing more feature channels: n=6, position(P); n=12, position + velocity(PV); n=18, position + velocity + acceleration (PVA). If the adjacent finger information included, the dimension of feature vector n doubled.

C. Real-time Prediction System

The prediction system used in this study is similar to the previous description. The proposed architecture is shown in

Fig. 5. Each sampling time is individually preprocessed and used as input for feature extraction. Each sampling time has its features extracted independently. After the multimodal fusion, the ideal feature structure is confirmed. The prediction system should guarantee the time cost for making a prediction is less than the time from the initial to final gesture position. When the hand starts moving from the initial position to complete the rock gesture, the raw data collected by data glove were used as input to the feature extraction algorithm, which generates multimodal fusion feature vectors to be employed in the prediction model. To avoid the outliers, each of the five features form a group. A boosting method is used to obtain a middle prediction gesture as the output of each feature group.

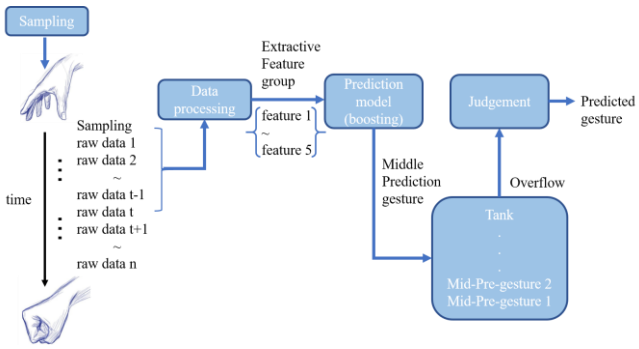

Fig. 5. Prediction system architecture

The middle prediction gesture is placed on the bottom of tank, where the capability of it depends on the tank length. When the tank is full, all of the middle prediction gestures in the tank are sent for judgement. After the process of judgement, we obtain the predicted gesture. The judgement algorithm is as follows. The parameter tank length is adjusted to eliminate the influence of the interim period before the extractive feature can represent the typical changing characteristics of the gesture.

Algorithm 1 Judgement algorithm

Input: Mid-Pre-gestures in Tank
Output: prediction gesture
While Tank full ==True
Get the prediction gesture (class) with **maximum number** in **Mid-Pre-gesture**
If prediction gesture==start gesture
Clean all the Mid-Pre-gesture in Tank
Else
Return prediction gesture

To describe the prediction model, a multilayer perceptron (MLP) neural network and multiclass support vector machine (SVM) are combined to form the prediction algorithm.

1. Multilayer perceptron

A n-hidden-layer Multilayer perceptron(MLP) is used with input vector $x=[1,x_1,x_2,x_3,...,x_m]$ and weight $W=[\omega_0,\omega_1,\omega_2,\omega_3,...,\omega_n]$ in first hidden layer, where m is the size of input vector, n is the neural number of hidden layer and ω_0 is the bias. The output vector $f(x)$ in matrix notation:

$$f(x)=G(s_n(W^{(n)}...(s_2(W^{(2)}(s_1(W^{(1)}x))))))) \quad (7)$$

Activation functions of hidden layer use the function which accelerate the computation speed of forward propagation, is defined as follows:

$$\mathrm{Re}lu(a)=\max(0,a) \quad (8)$$

G as the *softmax* function to achieve the multi-class classification.

$$G(x)_i=\frac{\exp(\omega_i^T x)}{\sum_{k=1}^{n}\exp(\omega_i^T x)} \quad (9)$$

The coding method of output is *one-hot*. We use the cross-entropy cost function while training the parameters, y is the ideal output, a is the real output:

$$C=-\frac{1}{n}\sum[y\ln a+(1-y)\ln(1-a)] \quad (10)$$

2. Support vector machine

The objective function is:

$$\max(1/\|\omega\|)\ s.t., y_i(w^T x_i+b)\geq 1, i=1,...,n \quad (11)$$

Using Lagrange multiplier α to transfer the question into dual variable optimizer question. Merge the constraint condition into the objective function, where w and b are the weight and bias respectively:

$$L(w,b,\alpha)=\frac{1}{2}\|w\|^2-\sum_{i=1}^{n}\alpha_i(y_i(w^T x_i+b)-1) \quad (12)$$

To solve the nonlinear task and simplify the calculation, Kernel function is used:

$$\max\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j\kappa(x_i,x_j) \quad (13)$$
$$s.t., \alpha_i\geq 0, i=1,...,n$$
$$\sum_{i=1}^{n}\alpha_i y_i=0$$

There are different kernel function kernel function $\kappa(x_i,x_j)$, and Eq.14 is Gaussian RBF kernel (*rbf*), while Eq.15 is polynomial kernel function (*poly*).

$$\kappa(x_i,x_j)=\exp(-\frac{\|x_i-x_j\|^2}{2\sigma^2}) \quad (14)$$

$$\kappa(x_i,x_j)=((x_i*x_j)+1)^d \quad (15)$$

To achieve the multiclass prediction, we try "one-verse-rest" (*ovo*) and "one-verse-rest" (*ovr*) methods [16]. We also explore different kinds of kernels of SVM to combine the MLP as the combined neural network to obtain prediction model in experiments [17]. Following are the combining algorithm.

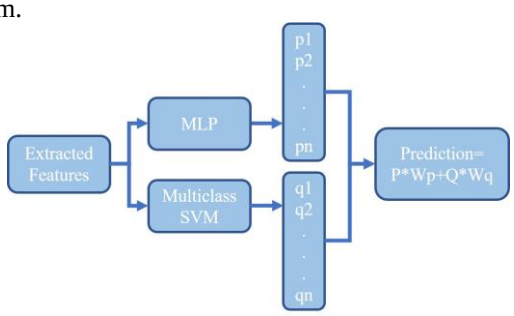

Fig. 6. Structure of combining algorithm

P(p1…pn) are confidence values extracted from the MLP, which represent the probability of the features belonging to each class of gesture, while *Q(q1…qn)* is extracted from the SVM, where *Wp + Wq = 1*, which represent the weight of MLP and SVM separately.

III. Experiments and Results

The accuracy and effectiveness experiments are executed in using the Flex-gesture dataset and the enlarged n-dimension data. We use two machine learning algorithms, SVM and MLP, for the classification model. For SVM, we use the Gaussian RBF kernel and 'one vs rest' method. The sigma value of RBF kernel is 0.1. For MLP, we use three hidden layers MLP with 50, 25 and 18 hidden units in each hidden layer. After determining the training model, we tested three feature vectors based on the criterion of computational cost and accuracy, and determine the best one. The feature vectors were position(P), position + velocity(PV) and position + velocity + acceleration (PVA). The output of the two models were identical, with 17 units representing the 17 classes of hand gestures.

The models trained with different kind of feature vector, different machine learning algorithm, with single finger feature was tested by the test dataset, Fig. 7. summarizes the prediction performance of each feature vectors.

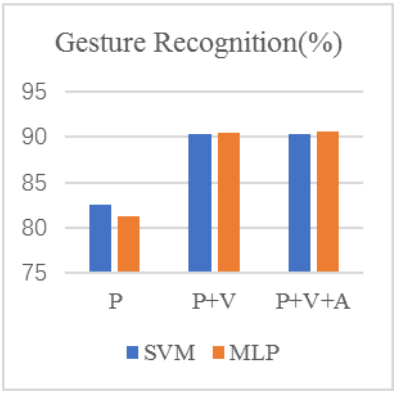

Fig. 7. Gesture prediction experiment results with single finger feature

Fig. 7 illustrates that PV feature significantly improves the accuracy of prediction from 82.47% to 90.31% with middle computational time based on SVM. The result shows that the velocity information is a very important factor to predict the gesture, but when the acceleration information is added to the feature, the improvement was limited, from 90.31% to 90.33% with SVM. In addition, the computational cost of the PVA feature vector (18 dimensions) was 1.5 times larger than the PV feature vector (12 dimensions), which significantly increases the computation cost, but slows down the prediction.

We repeated the experiment by adding the adjacent finger information to the feature vector, which double the dimension. Fig. 8 summarizes the prediction performance of each feature vectors.

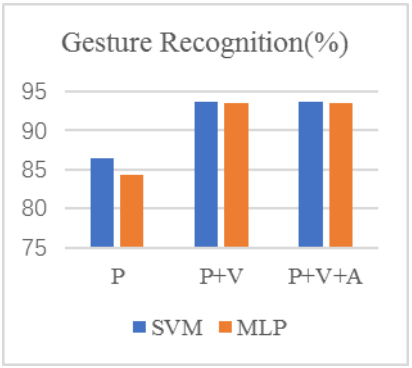

Fig. 8. Gesture prediction experiment results with adjacent finger feature

Fig. 8 shows that adding the adjacent finger information significantly improves the accuracy, when the SVM+PV features are chosen, it improves the accuracy from 90.31% to 93.61%. Therefore, it is helpful in distinguishing between different types of interactions between adjacent fingertips. All in all, each finger's position and palm's position reflect the spatial features, while temporal features are reflected by velocity and acceleration. The combination of these three different feature channels form a multimodal fusion. Finally, we adopt the adjacent PV as the ideal feature vector.

To select the optimal prediction model, we combine the different MLP and multiclass SVM algorithms to explore the ideal prediction model, using the time cost and accuracy as the evaluating indicator to adjust the parameter of the prediction model.

In experiment, the kernel of SVM are polynomial kernel (*poly*) and Gaussian RBF kernel (*rbf*), where one verse one (*ovo*) and one verse rest (*ovr*) methods are used. The structure of MLP are three hidden layers and the size of hidden layers are 50, 25, 18 respectively (MLP(3)), and four hidden layers with the size of hidden layers are 50, 25, 18, 16 respectively (MLP(4)) . Time represent the prediction time of each moment.

TABLE I. EXPERIMENT RESULTS FOR DIFFERENT PREDICTION MODEL

Prediction model	*Accuracy(%)*	*Time(ms)*
MLP(3)	*93.45*	*0.2315*
MLP(4)	*94.52*	*0.2329*
SVM ovr rbf	*95.20*	*0.0444*
SVM ovo rbf	*95.18*	*0.0444*
SVM ovr poly	*94.68*	***0.0272***
SVM ovo poly	*94.67*	***0.0272***
SVM ovo rbf + MLP(3)	*96.52*	*0.2315*
SVM ovr rbf + MLP(3)	*97.21*	*0.2315*
SVM ovo poly + MLP(3)	*96.50*	*0.2315*
SVM ovr poly + MLP(3)	*96.51*	*0.2315*
SVM ovo rbf + MLP(4)	***98.29***	*0.2329*
SVM ovr rbf + MLP(4)	*98.17*	*0.2329*
SVM ovo poly + MLP(4)	*97.22*	*0.2329*
SVM ovr poly + MLP(4)	*97.21*	*0.2329*

Table I illustrates that the prediction speed of both RBF kernel SVM and polynomial kernel SVM are very fast with cost time 0.0444 millisecond (*ms*) and 0.0272 *ms* respectively. The prediction speed of MLP is also fast with cost time 0.2315 *ms* (MLP (3)) and 0.2329 *ms* (MLP (4)). The combined prediction model has nearly the same speed with MLP only. The accuracy of only SVM, 95.20% with RBF kernel and *ovr*

method, is better than only MLP, 94.52% with 4 hidden layers. After combining different multiclass SVM and MLP algorithms, the accuracy significantly improved, where the SVM *ovo*, *rbf* + MLP (4) combined model had the highest prediction accuracy 98.29%. Because time cost is quite short which have little influence of prediction. Finally, we choose the highest accuracy combination, SVM *ovo*, *rbf* + MLP (4), as the prediction model.

To test the feasibility and performance of the prediction system proposed in this paper. We developed a graphical interface based on Python for gesture prediction. By measuring the bend of flex sensors, the embedded hardware platform makes the data acquisition, and transmits the data to the host computer wirelessly for data preprocessing based on multimodal fusion, converted into a set of features on the behave of the hand movements, then the prediction system uses extracted features to predict the gesture before the action ends. Fig. 9, Fig. 10, Fig. 11 are the gesture prediction example results in the experiments.

Fig. 9. Gesture "like" real-time prediction

Fig. 10. Gesture "rock" real-time prediction

Fig. 11. Gesture "scissor" real-time prediction

IV. DISCUSSION

Data processing methods form extraction features from the raw flex data. To combine both spatial and temporal channels, we extract the hidden features from the data, such as the velocity and acceleration. And the adjacent finger distance, which distinguishes between the different types of interactions between adjacent fingertips. Experiment results show that they are useful features to predict the next gesture position in next time. When evaluating the model using the PVA parameter in the flex-gesture dataset, our prediction model worked no better than the system using PV parameters. However, the computational cost of the PVA feature vector (18 dimensions) was 1.5 times larger than the PV feature vector (12 dimensions). In consideration of computational cost and accuracy, we chose the adjacent PV parameters as our ideal feature vector.

From the results of the prediction model experiment, cost time and accuracy are considered as the optimization index. The cost time of prediction modal play an important role in entire prediction time of prediction system. The experiment results show the cost time of prediction model can be neglect compared with the gesture time from initial to the end, which approximately 5 to 6 seconds. Thus the accuracy is the most concern to select the combination of prediction model. SVM *ovo*, *rbf* + MLP (4) combined model had the highest prediction accuracy 98.29% with 0.2329 *ms* cost time.

V. CONCLUSION

The proposed system uses three modules based on data glove to predict incomplete gestures before the data of entire process has been captured: sensors data collection, feature extraction based on multimodal fusion, and real-time gesture prediction. Data glove contains six flex sensors, an Arduino board, and a wireless module. Biological characteristics of hand muscle distribution guide the position and direction of flex sensors on each finger as well as palm, which collect exact spatial information. For data preprocessing, we perform data augmentation to obtain larger training datasets, and calculate the temporal information, bending velocity and acceleration, and the adjacent coupling information, adjacent distance between fingers. To avoid instability, the moving average method, a type of low-pass filter is used. The multimodal fusion method obtains the ideal feature vector. The experimental results show that adjacent PV feature vector has the highest accuracy. The gesture prediction algorithm guarantees the fast prediction using an SVM and MLP-combined model. The combination method was evaluated, and the results showed that SVM *ovo*, *rbf* and MLP-combined model has 98.29% prediction accuracy, consuming only 0.2329 *ms*. The feasibility experiment with graphical interface and host computer shows that the prediction system in data glove can make fast and accuracy prediction application.

The proposed structure has some limitations because of the lack of time dimension. In the future, we plan to try different deep learning algorithms which can process the data in time domain, combining time dimension into multimodal fusion. Using more powerful hardware may accelerate the speed of predicting, besides, obtain more reliable results.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under Grant No. 51475305 and 61473192.

REFERENCES

[1] Bodiroza, S. , Doisy, G. , and Hafner, V, "Position-invariant, real-time gesture recognition based on dynamic time warping," In: Human-Robot Interaction (HRI), 8th ACM/IEEE International Conference on, pp. 87–88, 2013.

[2] Lee, S.-W. "Automatic gesture recognition for intelligent human-robot interaction," In: Automatic Face and Gesture Recognition. 7th International Conference on, pp. 645–650, 2006.

[3] Liu, S. and Xiao, Q, "A signer-independent sign language recognition system based on the weighted knn/hmm," In: Intelligent Human-Machine Systems and Cyber- netics (IHMSC), 7th International Conference on, vol. 2, pp. 186–189, 2015.

[4] Mori, A. , Uchida, S. , Kurazume, R. , Taniguchi, R.-I. , Hasegawa, T. , and Sakoe, H, "Early recognition and prediction of gestures," In: Pattern Recognition. ICPR. 18th International Conference on, vol. 3, pp. 560–563, 2006.

[5] Solis, F., . Martinez, D., Espinoza, O., Toxqui, C. "Automatic Mexican Sign Language and Digits Recognition using Normalized Central Moments." Applications of Digital Image Processing, 2016.

[6] G. Marin, F. Dominio, and P. Zanuttigh, “Hand gesture recognition with jointly calibrated leap motion and depth sensor,” Multimedia Tools Appl., pp. 1–25, 2015.

[7] H. Cheng, L. Yang, and Z. Liu, “A survey on 3D hand gesture recognitions,” IEEE Trans. Circuits Syst. Video Technol., 2015.

[8] A. Kurakin, Z. Zhang, and Z. Liu, “A real time system for dynamic handgesture recognition with a depth sensor,” in Proc. 20th Eur. Conf. Signal Process. , 2012, pp. 1975–1979.

[9] Barros, P., N. T. Maciel-Junior, B. J. T. Fernandes, B. L. D. Bezerra and S. M. M. Fernandes, “A dynamic gesture recognition and prediction system using the convexity approach,” Computer Vision and Image Understanding 155: 139-149, 2017.

[10] Preetham, C., G. Ramakrishnan, S. Kumar, A. Tamse, and N. Krishnapura. “Hand Talk-Implementation of a Gesture Recognizing Glove,” 2013 Texas Instruments India Educators' Conference (Tiiec 2013): 328-331, 2013.

[11] Jadhav, A. J., and M. P. Joshi, “AVR based embedded system for speech impaired people,” 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (Icacdot): 844-848, 2016.

[12] Bhaskaran, A. K., A. G. Nair, D. K. Ram, K. Ananthanarayanan, and H. R. N. Vardhan. “Smart Gloves for Hand Gesture Recognition Sign Language to Speech Conversion System,” International Conference on Robotics and Automation for Humanitarian Applications (Raha): 226-231, 2016.

[13] Neverova, N., C. Wolf, G. Taylor and F. Nebout. “ModDrop: Adaptive Multi-Modal Gesture Recognition,” Ieee Transactions on Pattern Analysis and Machine Intelligence 38(8): 1692-1706, 2016.

[14] Wanqing Li, Zhengyou Zhang, and Zicheng Liu, “Expandable data-driven graphical modeling of human actions based on salient postures,” IEEE Trans. Circuits Syst. Video Techn., vol. 18, no. 11, pp. 1499–1510, 2008.

[15] Lu, W., Z. Tong and J. Chu. “Dynamic Hand Gesture Recognition With Leap Motion Controller,” IEEE Signal Processing Letters 23(9): 1188-1192, 2016.

[16] N. Cristianini and J. Taylor. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge UP, 2000.

[17] Zhongchang Sun, Huadong Guo, Xinwu Li, Linlin Lu, and Xiaoping Du, "Estimating urban impervious surfaces from Landsat-5 TM imagery using multilayer perceptron neural network and support vector machine,"Journal of Applied Remote Sensing, 2011.