Novel Feature Extraction and Classification Technique for Sensor-Based Continuous Arabic Sign Language Recognition

Mohammed Tuffaha[1], Tamer Shanableh[1] and Khaled Assaleh[2]

[1] American University of Sharjah, Department of Computer Science and Engineering, Sharjah, UAE
[2]American University of Sharjah, Department of Electrical Engineering, Sharjah, UAE
b00054842@aus.edu, tshanableh@aus.edu,
kassaleh@aus.edu

Abstract. This paper proposes a novel approach to continuous Arabic Sign Language recognition. We use a dataset which contains 40 sentences composed from 80 sign language words. The dataset is collected using sensor-based gloves. We propose a novel set of features suitable for sensor readings based on covariance, smoothness, entropy and uniformity. We also propose a novel classification approach based on a modified polynomial classifier suitable for sequential data. The proposed classification scheme is modified to take into account the context of the feature vectors prior to classification. This is achieved through the filtering of predicted class labels using median and mode filtering. The proposed work is compared against a vision-based solution. The proposed solution is found to outperform the vision-based solution as it yields an improved sentence recognition rate of 85%.

Keywords: Sign language recognition; feature extraction; sensor-based gloves; pattern classification.

1 Introduction

Sign language is the term used to describe the language that the deaf community uses to communicate together or with the hearing society. Sign language recognition systems are used to translate sign language into text or speech. Sign Language recognition systems have been developed for many languages including but not limited to English, Chinese, Korean and Arabic. Sign language recognition systems are divided into 2 categories based on data collection:

- Vision-based systems: Data is collected using one or more cameras. Typically such systems require high computational complexity and are not too accurate. An example of which is reported in [1].
- Glove-based systems: Data is collected using sensor-based gloves, an example of which is reported in [2]. It is more accurate than vision-based systems and is not affected by background motion, colors and light intensity.

adfa, p. 1, 2011.

https://doi.org/10.1007/978-3-319-26561-2_35

Moreover, sign language recognition systems can be used for recognizing sign language alphabet, words or sentences. The latter is used in this paper and it is also referred to as contentious sign language recognition.
Three sensor-based gloves were used for Arabic sign language recognition [3]; the PowerGlove, DT Data Gloves and CyberGlove. With the use of PowerGlove, the work in [4] developed an Arabic sign language recognition system using Support Vector Machine (SVM) classifier. While in [2] it was reported that DG5-VHand 2.0 data gloves is suitable for Sign Language Recognition as it contains flex sensors and accelerometers without the need for motion detectors. The classification system is based on a method of accumulated differences to eliminate the temporal dependencies in the data with low-complexity. This word-based system worked on two modes; user dependent mode produced accuracy of 95.3% and user independent mode achieved 92.5% accuracy. Using the same gloves, a user-dependent recognition system was proposed by [5], the system works on continuous Arabic Sign Language (ArSL).
In [6], the CyberGlove and Flock of birds 3D motion tracker were used for American Sign Language (ASL) recognition using neural networks. The algorithm is designed to recognize one-handed words. The system is trained and tested on a set of 50 words for single and multiple users. In [7], CyberGlove are also used for continuous single handed American Sign Language (ASL) recognition. Classification was done using a two layer Conditional Random Field (CRF), Support Vector Machines (SVM) and Bayesian network (BN). CyberGlove is also used in [8] to develop a user independent two handed Chinese Sign Language recognition for isolated and continuous signs.
In this work we propose a sentence-based Arabic sign language recognition system using sensor-based gloves. Both hands are used for collecting data from sign language sentences ranging from 3 to 7 words each. We propose a novel set of features suitable for sensor-based sign language data. We also propose a novel alternative for existing classification techniques based on polynomial classifier.

2 The Dataset

In this section, the dataset used for training and testing will be described. The current work makes use of data collected earlier by [5]. The collected data made use of the DG5-VHand 2.0 data gloves that give 8 sensor readings per hand; 5 flex sensors for each finger and 3 3D-accelerometers.
The dataset used was created in collaboration with Sharjah City for Humanitarian Services [9]. The dataset consists of 40 sentences built from 80 words. Each sentence is composed of 3 to 7 words. The sentences are the same as the ones reported in [10]. Each of the 40 sentences was repeated 10 times performed by a single user. Again, a sequence of sensor readings make up a sign language word and a sequence of words make up a sign language sentence.

3 Data Collection and Labeling

A sensor based approach is to be used and compared against previous work to demonstrate results. The process start by collecting two sets of data one from sensory

gloves and one from a camera. The camera input is used to label the sensory input manually.
Below is a descriptive diagram showing the process of data collection and labeling as used in [5]. The process starts by signing a new sentence, from which 2 attributes are acquired; sensor readings and record video. The sensor readings are manually labeled from the video and store the labels in a database. This process is illustrated in Figure 1.

4 Proposed Feature Extraction

This section describes the proposed process by which meaningful features are extracted from a set of raw data. The feature extraction is performed on a window of sensor readings. This is important to put each sensor reading in its right context. In this case the raw data is 16 sensors from 2 DG5-VHand 2.0 data gloves divided as follows: 2×5 flex sensors for each finger and 2×3 3D accelerometer sensors to determine the rotation and position of the hand in 3D space.
Consider vector f_i containing the 16 sensor readings as one vector at a single point in time. The matrix of readings F_i will be expressed as $F_i = [f_1 \ f_2 \ \dots \ f_N]^T$ where N is the number of vectors in a single window depending on time variant of the window and the number of readings taking per second. Typically 30 sensor readings per seconds are captured.
The raw data proved to contain some noise and are not very representative of the actual input so other statistical data are obtained from the feature vectors and added to make a new enriched FVs. In the beginning, mean and standard deviation were computed. Mean (μ) provides the average which eliminates to some extent the noise from the original feature vectors shown in the equation below.

$$\mu = \frac{1}{T}\sum_{k=1}^{T} f_k \tag{1}$$

While the standard deviation measures the dispersion from the average sample. In this case, an estimate of the standard deviation called sample standard deviation (s) is used to reduce complexity, it is calculated using the equation below.

$$s = \left(\frac{1}{T-1}\sum_{k=1}^{T}(f_k - \mu)^2\right)^{1/2} \tag{2}$$

For the sign language data, this is done on a window of feature vectors, where the size of the window is varied via trial and error to find achieve highest performance. For a predefined window size, the mean and standard deviation calculations are show below.

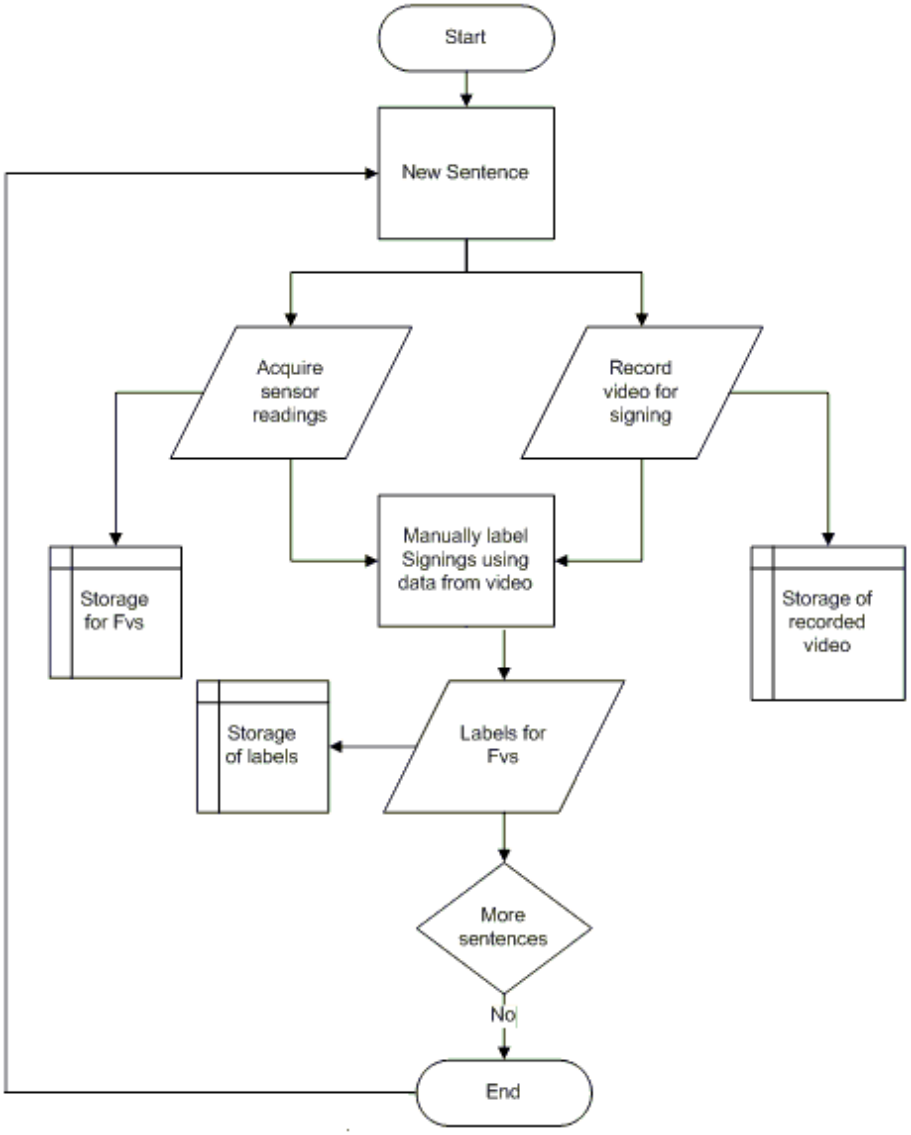

Fig. 1. Flowchart of data collection and labeling process.

$$\mu_i = \frac{1}{\omega} \sum_{k=i-\frac{\omega-1}{2}}^{i+\frac{\omega-1}{2}} f_k \tag{3}$$

$$s_i = \left(\frac{1}{\omega - 1} \sum_{k=i-\frac{\omega-1}{2}}^{i+\frac{\omega-1}{2}} (f_k - \mu)^2 \right)^{1/2} \quad (4)$$

Where ω is an odd number denoting the window size and i is the current feature from a set of features.

Then other statistical features are added to further enrich the feature vectors such as the covariance, the entropy and the uniformity.

The covariance shows how much the features change with respect to each other. Mathematically, covariance is calculated using the equation below.

$$cov(x, y) = \frac{\sum_{i=1}^{n}(x_i - \mu_x)(y_i - \mu_y)}{n - 1} \quad (5)$$

In this work we compute the covariance of sensor readings for a window of size ω. We use the upper triangular values of the covariance matrix as feature variables.

Where x is the independent variable, y is the dependent variable, n is the number of points in the sample, μ_x is the mean of variable x and μ_y is the mean of variable y.

The final feature vector consists of 200 values divided as follows: 16 raw sensor readings, 16 values for each of the window-based mean, standard deviation, entropy and uniformity and 120 values for the window-based covariance.

Since the data features have different scales, normalization is necessary to set a common range before classification for it to be successful. In this case, z-score normalization is used, the end result will have a mean of zero and unit variance.

5 Proposed Classification Solution

The proposed classification solution is based on the Polynomial classifier which was successfully used for classifying isolated sign language words [1]. Hence we start with a brief review of the polynomial classifier.

A Polynomial classifier is a supervised classifier technique which nonlinearly expands a sequence of input vectors to a higher dimension and maps them to a desired class labels.

Training a P^{th} order polynomial network is done in two stages. Stage one is expanding the training feature vectors through polynomial expansion. Stage two is linearly mapping the polynomial-expanded vectors to class labels by minimizing an objective criterion. Polynomial expansion of an M-dimensional feature vector $\mathbf{x} = [x_1 \ x_2 \ ... \ x_M]$ is achieved by combining the vector elements with multipliers to form a set of basis functions, **p(x).** The elements of **p(x)** are the monomials of the form $\prod_{j=1}^{M} x_j^{k_j}$, where

k_j is a positive integer, and $0 \le \sum_{j=1}^{M} k_j \le P$. For class i the sequence of feature vectors $\mathbf{X}_i = [\mathbf{x}_{i,1}\ \mathbf{x}_{i,2}\ \cdots, \mathbf{x}_{i,N_i}]^T$ is expanded into:

$$\mathbf{V}_i = [\mathbf{p}(\mathbf{x}_{i,1}) \quad \mathbf{p}(\mathbf{x}_{i,2}) \quad \cdots \quad \mathbf{p}(\mathbf{x}_{i,N_i})]^T \tag{6}$$

Where $\mathbf{X}_i$ is a $N_i \times M$ matrix and $\mathbf{V}_i$ is a $N_i \times O_{M,\mathrm{p}}$ matrix.

Expanding all the training feature vectors results in a global matrix for all K classes obtained by concatenating all the individual $\mathbf{V}_i$ matrices such that $\mathbf{V} = [\mathbf{V}_1 \quad \mathbf{V}_2 \quad \cdots \quad \mathbf{V}_K]^T$.

For each class i, the training objective is to find an optimum weight vector obtained by minimizing the distance between the class labels $\mathbf{y}_i$ and a linear combination of the polynomial expansion of the training feature vectors $\mathbf{V}\ \mathbf{w}_i$ such that

$$\mathbf{w}_i^{opt} = \arg\min_{\mathbf{w}_i} \left\| \mathbf{V}\ \mathbf{w}_i - \mathbf{y}_i \right\|_p \tag{7}$$

The class labels for the i^{th} class, $\mathbf{y}_i$, is a column vector comprised of ones and zeros such as $\mathbf{y}_i = [\ \mathbf{0}_{N_1}, \mathbf{0}_{N_2}, \ldots, \mathbf{0}_{N_{i-1}}, \mathbf{1}_{N_i}, \mathbf{0}_{N_{i+1}}, \ldots, \mathbf{0}_{N_k}\ ]^{\mathrm{T}}$.

In the identification stage we are given a sequence of N_c feature vectors $\mathbf{X}_c$ and we are required to determine its class c as one of the enrolled classes in the set $\{1,2,\cdots,K\}$. This is done by two steps: first, expand $\mathbf{X}_c$ into its polynomial basis terms $\mathbf{V}_c = [\mathbf{p}(\mathbf{x}_{c,1}) \quad \mathbf{p}(\mathbf{x}_{c,2}) \quad \cdots \quad \mathbf{p}(\mathbf{x}_{c,N_c})]^T$, and second, evaluate the output sequences against all K models $\left\{\mathbf{w}_i^{opt}\right\}$ to obtain a set of score sequences $\{\mathbf{s}_i\}$ such as

$$\mathbf{s}_i = \mathbf{V}_c \mathbf{w}_i^{opt} \tag{8}$$

The elements of the score sequence $\mathbf{s}_i$ represent the individual scores of each feature vector in the vector sequence $\mathbf{X}_c$. The class of the sequence $\mathbf{X}_c$ is determined by maximizing $\mathbf{s}_i$ such as

$$c = \arg\max_i (\mathbf{s}_i) \tag{9}$$

The proposed classifier is based on the above polynomial classifier of P^{th} order. Each feature vector is labeled according to the sign language word it belongs to. First, the classification weights vector is generated and then multiplied by the individual feature vectors to find the corresponding class label. We propose to use the context of the predicted label before arriving to the final classification result. We examine the predicted labels of the surrounding feature vectors and use a majority vote to decide on the class label of the current feature vector. Moreover, in Equation (12) above, instead

of computing the max score, we compute the first 3 maximum scores for each feature vector in the context window and then apply the majority vote. Figure 2 shows a block diagram for the classification process.

6 Experimental Results

In our dataset, the individual words are labeled by giving all feature vectors making up a word the same label. That is, each feature vector is labeled separately. There is a total of 82 classes where 80 are for the words and 2 are for the start and end of the sentences. Following the experimental setup of [5], the data is divided into 70% training and 30% testing in a round robin fashion.

In the results to follow, the sentences are considered correctly classified if all the words of the sentence are recognized successfully. Hence, the recognition rate is calculated by dividing the number of correctly recognized sentences by the total number of sentences. The word recognition rate on the other hand is found using the equation below [11].

$$Rate_{word} = 1 - \frac{D + S + I}{N} \tag{10}$$

Where D is the number of deleted words, I is the number of inserted words, S is the number of substituted words, and N is the total number of the words.

We use three classification measures. The first is the accuracy of classifying each feature vector individually, we refer to it as "class (%)". The second is the accuracy of classifying words based on the equation above, which is a sequences of feature vectors, we refer to it as "word (%)". The third and last is the accuracy of classifying each sentence correctly, which is a sequence of words, we refer to this as "sentence (%)". The tests are run on a Windows 7 machine with intel-i7 4790k 3.5GHz processor (quad core) and RAM of 16GB.

In Table 1, we present the classification results using raw sensor features with window-based mean and standard deviation (total of 48 features). Polynomial classifier was implemented from 2nd to 7th order. In this experiment the FVs window ω is 43 and context width is 32.

Table 1. Classification results using raw sensor features with window-based mean and standard deviation

Order	Class (%)	Word (%)	Sentence (%)	Time
2nd	66.94	44.38	33.89	13s
6th (peak)	81.6	70.85	63.33	39s

The same results are repeated with the addition of the upper diagonal window-based covariance of the features (total of 168 features). The results are presented in Table 2. In this experiment the FVs window ω is 63 and context width is 24.

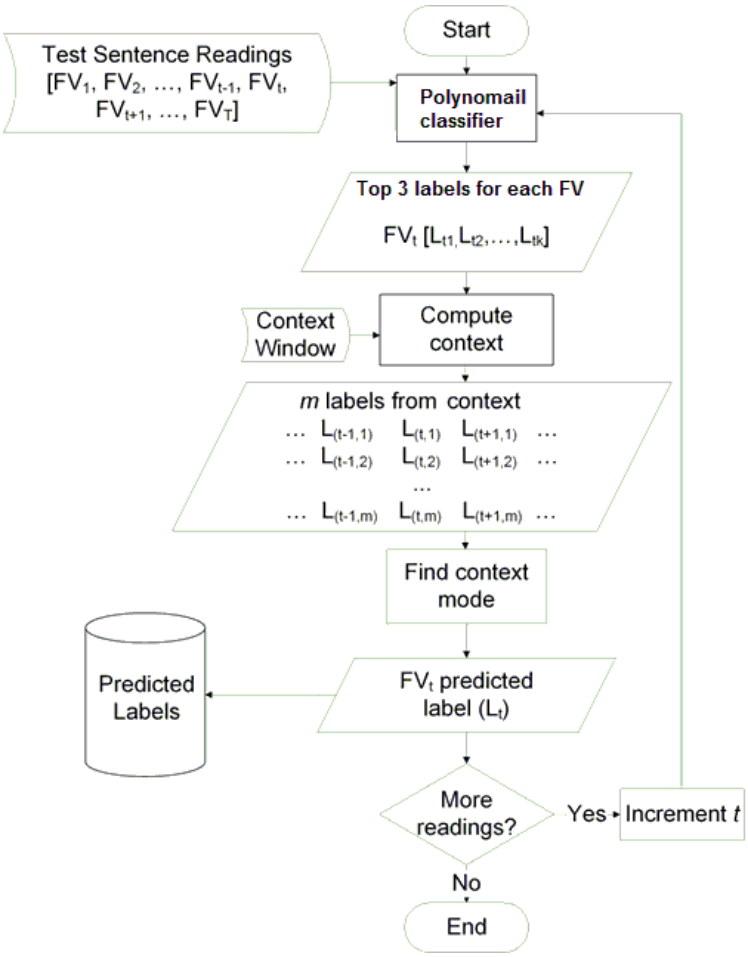

Fig. 2. Flowchart of the proposed classification system

Table 2. Classification results using the features in Table 1 and the upper diagonal window-based covariance.

Order	Class (%)	Word (%)	Sentence (%)	Time
2nd	82.72	77.06	71.11	62s
5th (peak)	87.48	86.67	82.22	142s

The entropy and uniformity can also be added to the above feature set. This brings up the total features to 200 variables. The classification results of which are reported in Table 3. In this experiment the FVs window ω is 75 and context width is 22.

Table 3. Classification results using the features in Tables 1 & 2 and the entropy and uniformity of features

Order	Class (%)	Word (%)	Sentence (%)	Time
2nd	84.37	79.80	76.39	75s
5th (peak)	87.77	88.95	85.00	135s

The results in the above tables show that the proposed feature variables are suitable for sensor-based sign language recognition. The results also show that the proposed feature extraction and classification approach are not computationally expensive. For instance in Table 3, at a second order polynomial, it takes an average of 0.6 second to classify a sign language sentence.
Figure 3 plots the sentence classification rate as a function of the polynomial classification order from 2 to 6. It is shown that the classification rates peaks at the 5th order.

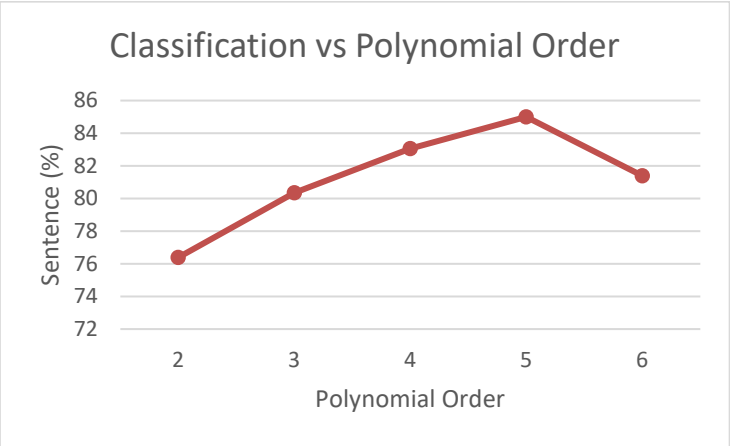

Fig. 3. Classification vs Polynomial Order

Lastly, the proposed sensor-based solution is compared to [10] in terms of classification accuracy. As mentioned, the work in [10] used the same set of sentences with similar experimental setup. Features are based on Discrete Cosine Transform of window-based accumulated image differences. Hidden Markov Models are used for classification. The results are show in Table 4.

Table 4. Classification rates of proposed and reviewed solutions.

	Proposed	**Reviewed [10]**
Sentence recognition rate	85%	75.6%

This result is expected because sensor-based data is more accurate than vision-based data. Sensor readings are specific to hand movements whereas in sign language videos hand movements need to be segmented out. Segmentation techniques are typically not accurate and therefore the features are not an exact representation of the sign language sentences.

7 Conclusion

We proposed to modify the polynomial classifier to work with sequential data. This is implemented using a window-based feature extraction approach and through

the use of statistical filtering of the predicted labels. We also proposed a new set of window-based features based on covariance, entropy, uniformity, smoothness and skewness. These features enhanced the classification accuracy. Lastly, we showed that the proposed system is computationally attractive and more accurate than existing vision-based solutions.

Acknowledgement

The authors gratefully acknowledge the American University of Sharjah for supporting this research through grant FRG14-2-26.

References

1. T. Shanableh and K. Assaleh, "User-independent recognition of Arabic sign language for facilitating communication with the deaf community," *Digital signal processing,* vol. 21, pp. 535-542, 2011.
2. K. Assaleh, T. Shanableh, and M. Zourob, "Low Complexity Classification System for Glove-Based Arabic Sign Language Recognition," in *Neural Information Processing*. vol. 7665, T. Huang, Z. Zeng, C. Li, and C. Leung, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 262-268.
3. M. Mohandes, M. Deriche and J. Liu, "Image-Based and Sensor-Based Approaches to Arabic Sign Language Recognition," *Human-Machine Systems, IEEE Transactions on,* vol. 44, pp. 551-557, 2014.
4. M. Mohandes, S. A-Buraiky, T. Halawani and S. Al-Baiyat, "Automation of the Arabic sign language recognition," in *Information and Communication Technologies: From Theory to Applications, 2004. Proceedings. 2004 International Conference on*, 2004, pp. 479-480.
5. N. Tubaiz, T. Shanableh and K. Assaleh, "Glove-Based Continuous Arabic Sign Language Recognition in User-Dependent Mode." *IEEE Transactions on Human-Machine Systems* PP, no.99 (2015), 1-8, doi: 10.1109/THMS.2015.2406692
6. C. Oz and M. C. Leu, "American Sign Language word recognition with a sensory glove using artificial neural networks," *Engineering applications of artificial intelligence,* vol. 24, pp. 1204-1213, 2011.
7. W. W. Kong and S. Ranganath, "Towards subject independent continuous sign language recognition: A segment and merge approach," *Pattern Recognition,* vol. 47, pp. 1294-1308, 3// 2014.
8. W. Gao, G. Fang, D. Zhao and Y. Chen, "A Chinese sign language recognition system based on SOFM/SRN/HMM," *Pattern Recognition,* vol. 37, pp. 2389-2402, 12// 2004.
9. "Sharjah City for Humanitarian Services (SCHS)." *http://www.schs.ae/indexs.aspx*
10. K. Assaleh, T. Shanableh, M. Fanaswala, F. Amin and H. Bajaj, "Continuous Arabic Sign Language Recognition in User Dependent Mode," *Journal of intelligent learning systems and applications,* vol. 2, pp. 19-27,
11. T. Starner, J. Weaver and A. Pentland, "Real-time American sign language recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 12, pp. 1371–1375, 1998.