International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 3 (2011) pp. 498-506

ArSLAT: Arabic Sign Language Alphabets Translator

Nashwa El-Bendary[1], Hossam M. Zawbaa[2], Mahmoud S. Daoud[2], Aboul Ella Hassanien[2], and Kazumi Nakamatsu[3]

[1] Arab Academy for Science,Technology, and Maritime Transport
23 Dr. ElSobki St., Dokki, 12311, Giza, Egypt
nashwa_m@aast.edu

[2] Faculty of Computers and Information, Cairo University
5 Ahmed Zewal St., Orman, Giza, Egypt
{aboitcairo, hossam.zawba3a}@gmail.com

[3] School of Human Science and Environment, University of Hyogo
1-1-12 Shinzaike-hon-cho, HIMEJI 670-0092, Japan
nakamatu@shse.u-hyogo.ac.jp

Abstract: This paper presents an automatic translation system for gestures of manual alphabets in the Arabic sign language. The proposed Arabic Sign Language Alphabets Translator (ArSLAT) system does not rely on using any gloves or visual markings to accomplish the recognition job. As an alternative, it deals with images of bare hands, which allows the user to interact with the system in a natural way. The proposed ArSLAT system consists of five main phases; pre-processing phase, best-frame detection phase, category detection phase, feature extraction phase, and classification phase. The used extracted features are translation, scale, and rotation invariant in order to make the system more flexible. Experiments revealed that the proposed ArSLAT system was able to recognize the Arabic alphabets with an accuracy of 91.3% and 83.7% using minimum distance classifier (MDC) and multilayer perceptron (MLP) classifier, respectively.

Keywords: Arabic Sign Language, Minimum Distance Classifier (MDC), Multilayer Perceptron (MLP) Classifier, Feature Extraction, Classification.

I. Introduction

Signing has always been part of human communications [1]. Newborns use gestures as a primary means of communication until their speech muscles are mature enough to articulate meaningful speech. For thousands of years, deaf people have created and used signs among themselves. These signs were the only form of communication available for many deaf people. Within the variety of cultures of deaf people all over the world, signing evolved to form complete and sophisticated languages.

Sign language is a form of manual communication and is one of the most natural ways of communication for most people in deaf community. There has been a re-surging interest in recognizing human hand gestures. The aim of the sign language recognition is to provide an accurate and convenient mechanism to transcribe sign gestures into meaningful text or speech so that communication between deaf and hearing society can easily be made. Hand gestures are spatio-temporally varying and hence the automatic gesture recognition turns out to be very challenging [2-6].

As in oral language, sign language is not universal; it varies according to the country, or even according to the regions. Sign language in the Arab World has recently been recognized and documented. Many efforts have been made to establish the sign language used in individual countries, including Jordan, Egypt, and the Gulf States, by trying to standardize the language and spread it among members of the deaf community and those concerned. Such efforts produced many sign languages, almost as many as Arabic-speaking countries, yet with the same sign alphabets [7]. Gestures used in Arabic Sign Language Alphabets are depicted in figure 1.

The significance of using hand gestures for communication becomes clearer when sign language is considered. Sign language is a collection of gestures, movements, postures, and facial expressions corresponding to letters and words in natural languages, so the sign language has more than one form because of its dependence on natural languages. The sign language is the fundamental communication method between people who suffer from hearing impairments. In order for an ordinary person to communicate with deaf people, an interpreter is usually needed to translate sign language into natural language and vice versa [3].

Human-Computer Interaction (HCI) is getting increasingly important as a result of the increasing significance of computer's influence on our lives [3]. Researchers are trying to make HCI faster, easier, and more natural. To achieve this, Human-to-Human Interaction techniques are being introduced into the field of Human-Computer Interaction. One of the richest Human-to-Human Interaction fields is the use of hand gestures in order to express ideas.

Figure 1. Arabic sign language alphabets

In the recent years, the idea of the computerized translator has become an attractive research area [3]. Existing HCI devices for hand gesture recognition fall into two categories: glove-based and vision-based systems [2]. The glove-based system relies on electromechanical devices that are used for data collection about the gestures [8-12]. The user has to wear cumbersome and inconvenient devices that make the interaction between the system and the user very complicated and less natural than the required HCI should be. In that case, the person must wear some sort of wired gloves that are interfaced with many sensors. Then, based on the readings of the sensors, the gesture of the hand can be recognized by a computer interfaced with the sensors. In order to get rid of the inconvenience and to increase the naturalness of HCI, the second category of HCI systems has been provided to overcome this problem.

Vision-based systems basically suggest using a set of video cameras, image processing, and artificial intelligence to recognize and interpret hand gestures [8]. Visual-based gesture recognition systems are further divided into two categories. The first one relies on using specially designed gloves with visual markers that help in determining hand postures [3]. However, using gloves and markers does not provide the naturalness required in such HCI systems. Besides, if colored gloves are used, the processing complexity is increased. As an alternative, the second kind of visual-based gesture recognition systems tries to achieve the ultimate convenience and naturalness by using images of bare hands to recognize gestures.

This paper presents the ArSLAT system, an Arabic Sign Language Alphabets Translator. The rest of this paper is organized as follows. Section II shows some of the related works concerning sign language translation systems. Section III describes the system architecture of the ArSLAT system and overviews its phases. In section IV, experiments and results are presented. Finally, section V summarizes conclusions and discusses future work.

II. Related Work

Signing has always been part of human communications. The use of gestures or sign is not tied to ethnicity, age, or gender [7]. In recent years, several research projects in developing sign language systems have been presented [13].

In [7], an Arabic Sign Language Translation Systems (ArSL-TS) model has been introduced. That presented model runs on mobile devices to develop an avatar based sign language translation system that allows users to translate Arabic text into Arabic Sign Language for the deaf on mobile devices such as Personal Digital Assistants (PDAs).

In [14], a virtual signer technology was described. The ITC (Independent Television Commission-UK) has specially made Televirtual to develop "Simon", the virtual signer in order to translate printed text - television captions - into sign language. The proposed model tried to solve some of the problems resulted from adding sign language to television programs. Also, authors discussed the language processing techniques and models that have been investigated for information communication in a transaction application in Post Offices, and for presentation of more general textual material in texts such as subtitles accompanying television programs.

The software proposed in [14] consists of two basic modules: linguistic translation from printed English into sign language, and virtual human animation. The animation software allows Simon to sign in real-time. A dictionary of signed words enables the system to look up the accompanying physical movement, facial expressions and body positions, which are stored as motion-capture date on a hard disk. The motion-capture data that includes hand, face and body information is applied to a highly detailed 3D graphic model of a virtual human. This model includes very realistic and accurate hand representations, developed within the project. Moreover, natural skin textures are applied to the hands and face of the model to create the maximum impression of subjective reality.

In [15], Data acquisition, feature extraction and classification methods employed for the analysis of sign language gestures have been examined. These were discussed with respect to issues such as modeling transitions between signs in continuous signing, modeling inflectional processes, signer independence, and adaptation.

Also, It has been stated that non-manual signals and grammatical processes, which result in systematic variations in sign appearance, are integral aspects of this communication but have received comparatively little attention in the literature. Works that attempt to analyze non-manual signals have been examined. Furthermore, issues related to integrating these signals with (hand) sign gestures and the overall progress toward a true test of sign recognition systems dealing with natural signing by native signers have been discussed.

Moreover, a summary of selected sign gesture recognition systems using sign-level classification has been presented in [15]. According to that summary, the two main approaches in sign gesture classification either employ a single classification stage, or represent the gesture as consisting of simultaneous components that are individually classified and then integrated together for sign-level classification. Another summary indicated the variety of classification schemes and features used under the two broad approaches. In each approach, methods that use both, direct-measure devices and vision are included.

In [16], an automatic Thai finger-spelling sign language translation system was developed using Fuzzy C-Means (FCM) and Scale Invariant Feature Transform (SIFT) algorithms. Key frames were collected from several subjects at different times of day and for several days. Also, testing Thai fingerspelling words video was collected from 4 subjects. The system achieves 79.90% and 51.17% correct alphabet translation and the correct word translation, respectively, with the SIFT threshold of 0.7 and 1 nearest neighbor prototype. However, when the number of nearest neighbor prototypes was increased to 3, the system yields higher percentages, 82.19% and 55.08% correct alphabet and correct word translation, respectively, at the same SIFT threshold.

Also, a system for automatic translation of static gestures of alphabets in American Sign Language (ASL) was developed in [17]. Three feature extraction methods and neural network were used to recognize signs. The developed system deals with images of bare hands, which allows the user to interact with the system in a natural way. An image is processed and converted to a feature vector that will be compared with the feature vectors of a training set of signs. The system is implemented and tested using data sets of number of samples of hand images for each signs. Three feature extraction methods are tested and best one is suggested with results obtained from Artificial Neural Network (ANN). The system is able to recognize selected ASL signs with the accuracy of 92.33%.

In [18], Authors discussed the development of a data-driven approach for an automatic machine translation (MT) system in order to translate spoken language text into signed languages (SLs). They aimed at improving the accessibility to airport information announcements for deaf and hard of hearing people. [18] also demonstrates the involvement of deaf members of the deaf community in Ireland in three areas, which are: the choice of a domain for automatic translation that has a practical use for the deaf community; the human translation of English text into Irish Sign Language (ISL) as well as advice on ISL grammar and linguistics; and the importance of native ISL signers as manual evaluators of our translated output.

The proposed system achieved a reasonable job of translating English into ISL with scores comparable to mainstream speech-to-speech systems. More than two thirds of the words produced are correct and almost 60% of the time the word order is also correct. Using the Marker Hypothesis to segment sentences improves both word error rate (WER) and position-independent word error rate (PER) scores, the latter by approximately 3% showing an increase in the number of correct words in the candidate translations. The results also show that sub-sentential chunking of the training data improves the translation.

III. ArSLAT: Arabic Sign Language Alphabets Translator System

The proposed Arabic Sign Language Alphabets Translator (ASLAT) system is composed of five main phases [19]: Pre-processing phase, Best-frame Detection phase, Category Detection phase, Feature Extraction phase, and finally Classification phase. Figure 2 depicts the structure of the ArSLAT system.

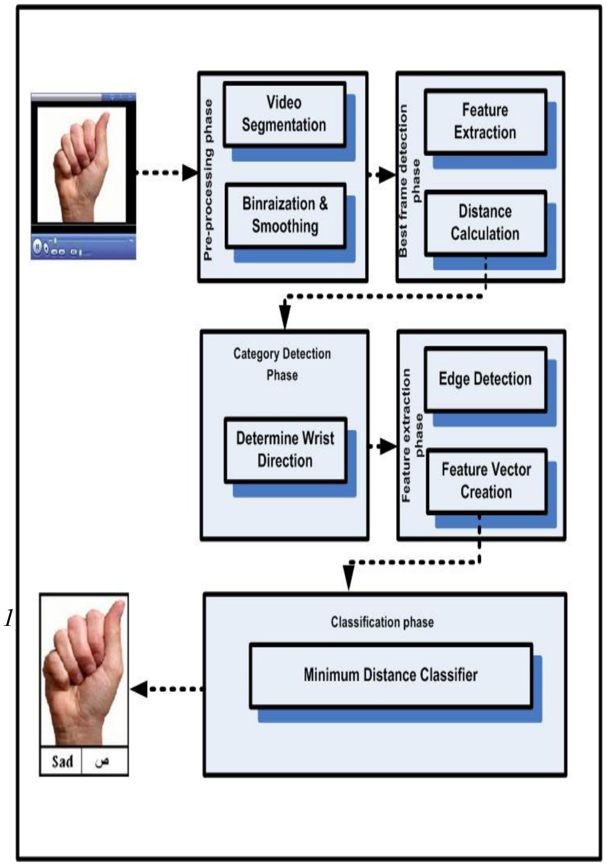
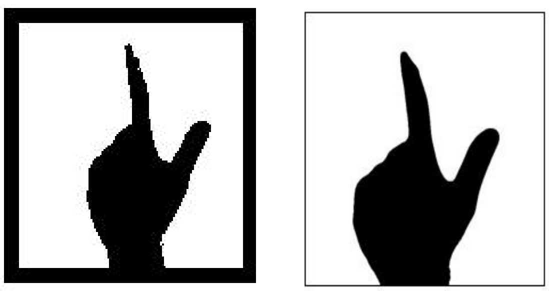


Figure 2. ArSLAT System architecture

Pre-processing phase receives, as an input, a video that contains the signed words to be translated into text, and prepare it to be ready for use in subsequent phases. In best-frame detection phase, the system detects the number of words that have been signed and the number of letters in each word then it takes snapshots of these letters. Category detection phase considers the Arabic sign language as three categories depending on the direction from which the hand wrist appears. Consequently, this phase focuses on specifying the category of all letters and accordingly helps the next phases to increase the accuracy of the recognition operation and minimize the processing time by reducing the matching operation. Feature extraction phase extracts features of each letter in order to be represented using these features, within the remaining phases of the system, according to its category. The extracted features are rotation, scale, and translation invariant. Finally, in classification phase, each unknown letter is being matched with all the known letters in the same category in the database and takes the nearest one to this letter and consequently, the system writes the result as text. The proposed interpreter system deals with the 30 Arabic sign language alphabets visually; using a recorded video contains the motion of the bare hands. The users are not required to wear any gloves or to use any devices to interact with the system [19].

A. Pre-processing phase

Firstly, a video that contains stream of signed words (gestures) to be translated is acquired. Then, the video enters the pre-processing phase where frames are captured from the video by applying a video segmentation technique that captures frames with a frame rate of 20 frames per second. Then, the captured frames are converted into binary format such that black pixels represent the hand gesture and white pixels represent the background or any object behind the hand as shown in Figure 3(a). Finally, smoothing is applied for each frame to remove noise and shadow as shown in Figure 3(b).



(a) Noise frame **(b)** Smoothed result

Figure 3. Pre-processing results

B. Best Frame Detection phase

For this phase, there is a stream of frames containing a word represented by a number of sign language gestures. Each letter in that word has been signed by a special hand view or hand sign. However, there are two issues here to be tackled. Firstly, how to know the number of letters of this word and the second one is how to detect only one frame that actually represents each letter.

The solution for these issues depends on logic instead of programming or mathematical algorithms. The idea depends on the way that the person says the word. The user signs the first letter then pauses for a certain time (1 second) followed by a change in the shape of the hand in order to sign the next letter then pauses again for the same pause time and so on. Consequently, this pause time is useful for detecting the letters by extracting features for each frame in the video with addition to comparing these features together in order to detect the number of letters and the frames representing them. In the case of detecting an empty frame (with no hand objects existed), this means that a new word will begin. Accordingly, the system takes this as an indicator for separating groups of letters in order to formulate words.

For each frame three features are calculated:

1. The distance from the top edge of the image to the first black pixel, which represents the first pixel of the hand.
2. The distance of the first black pixel from two-thirds of the right edge of the image.
3. The distance of the first black pixel from two-thirds of the left edge of the image.

These three distances represent the best position of the hand as depicted in figure 4.

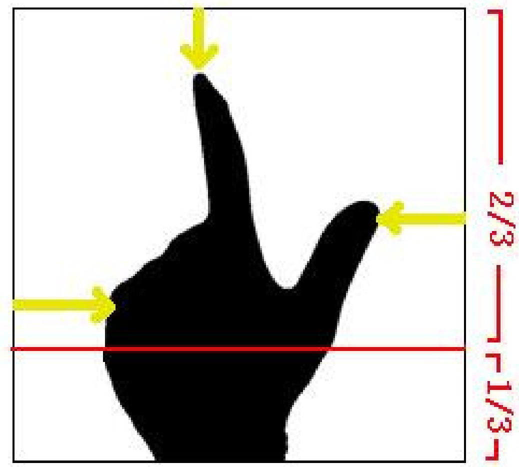


Figure 4. Indicator for the best position of the hand

These distances are the best indicator for the position of the hand and its movements through frames. Using the two-thirds of the left and right edges to avoid the wrist of the hand, the form of these features is a vector of length equals to three, which contains these three features as feature elements, as follows:

Feature vector = [*distance1* *distance2* *distance3*] (1)

Subsequently, the system calculates the distance between each frame and the following frame. One frame is skipped because the distance between any two consecutive frames is typically too small, and the pause time between signing letters will enhance the accuracy and decrease the processing time.

Finally, the system calculates the distance between feature vectors of frames through applying the Euclidian distance rule shown in equation (2).

$$D = \sqrt{\sum_{i=1}^{3}(x_i - y_i)^2} \qquad (2)$$

Where D is the distance between two feature vectors, x and y are the elements of the first and second vectors, respectively. Experimentally, it has been found that 250 is a good threshold to create ranges from these frames. Therefore, if any distance between two consecutive frames exceeds 250, the system will ignore this distance. Finally, the system will keep only ranges of frames with distances smaller than 250. Also, it has been found that the system selects ranges with length exceeds 4, the number of these ranges are mostly equal to the number of letters in the signed word. The middle frame for each of these ranges is selected for representing a letter as the middle is considered the safest one.

C. Category detection phase

This phase considers the Arabic sign language as three categories depending on the appearance direction of the hand wrist. When the user starts to make hand gestures in order to sign letters, some of these letters make the wrist appears from bottom-right like the letter "Waw" and other letters make the wrist appears from bottom-left like the letter "Dal". Moreover, some other letters make the wrist appears from the down-half like the letter "Ba". Therefore, category detection phase focuses on specifying the category of each letter. Consequently, this will help the following phases to increase accuracy of the recognition operation and minimize processing time by reducing the matching operation.

For determining the category of letters, the system cuts the image at each best-frame to resize the hand object. Then, the system checks the pixel with the maximum value for both horizontal and vertical axis (X-axes and Y-axes), which is marked with a circle in figure 5(a). If it is a black pixel (hand pixel), the letter belongs to the category with wrist appears from bottom-right. Otherwise, the system will check for the pixel with zero value for both horizontal and vertical axis (X-axes and Y-axes), which is marked with a circle in figure 5(b), if it is a black (hand pixel), the letter belongs to the category with wrist appears from bottom-left. If none of the previous situations exists, the letter belongs to the category with wrist appears at the middle of frame as in figure 5(c).

D. Feature extraction phase

The importance of feature extraction phase is to know the meaning of the letters and accordingly to understand the signed word. Feature extraction phase is divided into two stages, namely, edge-detection stage and feature-vector-creation stage. Therefore, each best-frame will go through edge-detection stage, which detects the frame edges using image processing filters, then returns a new frame containing only the contour pixels to make use of it in the following stage of the feature extraction phase that is feature-vector-creation stage. The edge-detection stage is shown in figure 6.

Before extracting features, the system specifies an essential point (orientation point) for each letter or frame. That orientation point is based on the category of each letter. Therefore, if wrist position is at the middle of the frame, the orientation point will be at the middle of the last row in the contour image. If wrist position is bottom-right, the orientation point will be at the most right column of the last row. However, if wrist position is bottom-left, the orientation point will be at the most left column of the last row, as shown in figure 7.

Using the determined orientation point, the system will extract different features for each category. For the first category with the wrist appears at the middle, the orientation point is at the middle of the last row in the contour and the total angle (Big$_{theta}$) that exists around the orientation point is equal to 180 degrees, so the minimum angle is zero and the maximum angle is 180 degrees, as shown in figure 8.

(a)

(b)

(c)

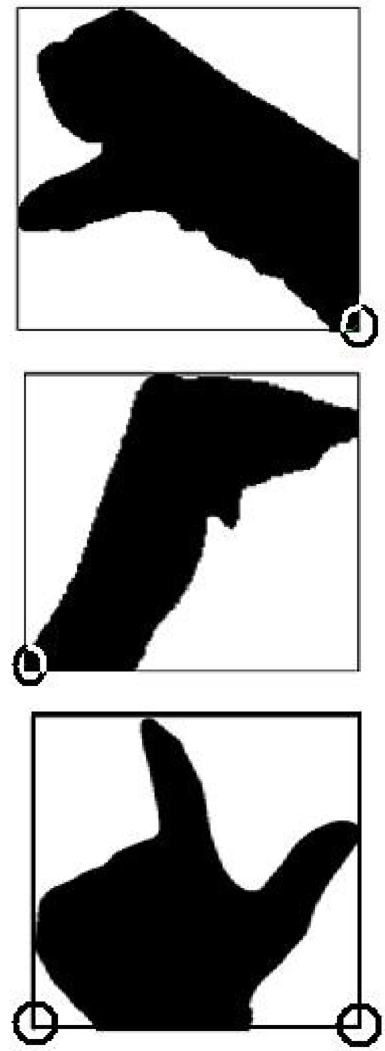


Figure 5. Determining the category of each letter

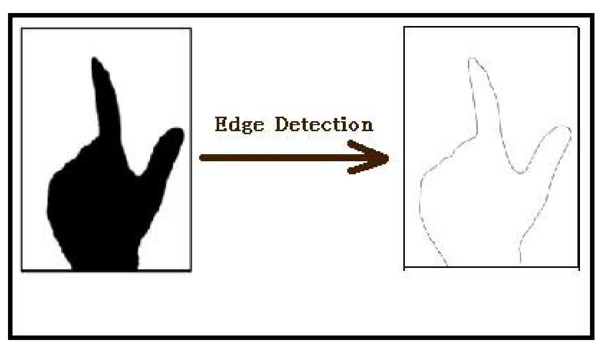


Figure 6. Edge-detection stage

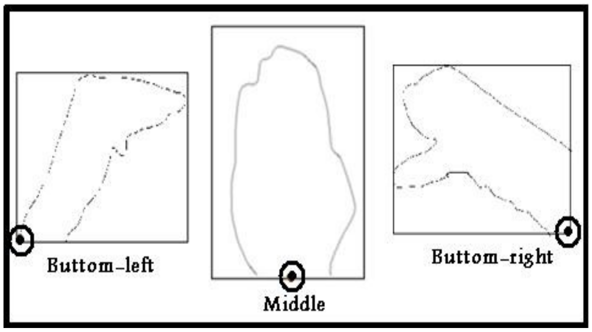


Figure 7. Location of orientation point

A feature element is a distance between the orientation point and a point P on the contour of the hand. Figure 9 shows the feature element, which is calculated by equation (3).

$$D_{Cb} = \sqrt{(b_x - c_x)^2 + (b_y - c_y)^2} \qquad (3)$$

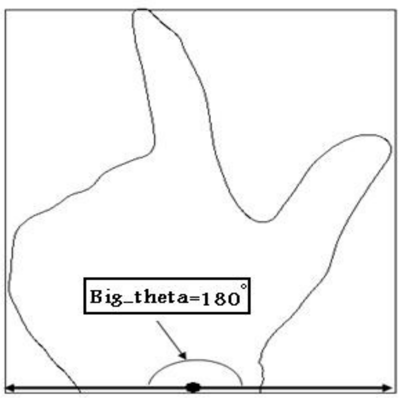


Figure 8. The first category: wrist appears at the middle and Big_{theta} =180°

The feature vector is a set of feature elements. However, by experiments, it has been obtained that 50 points equally spaced by certain angle ɵ (theta) are enough to detect the meaning of the letter with high accuracy. The theta angle ɵ (theta) calculated by equation (4)

$$\theta_{cb} = tan^{-1}\frac{(b_y - c_y)}{(b_x - c_x)} \qquad (4)$$

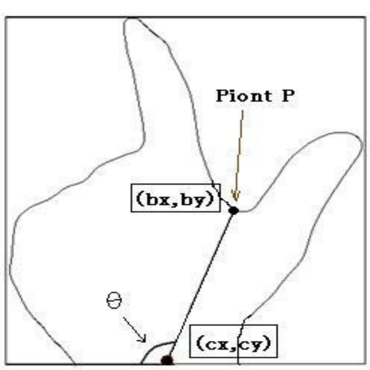


Figure 9. Feature element

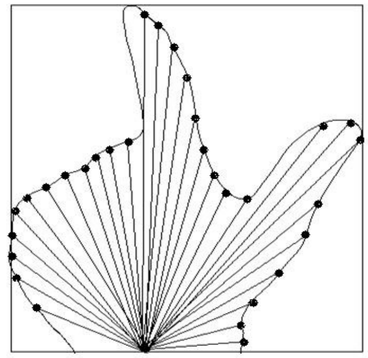


Figure 10. The fifty feature elements that make up the feature vector

Therefore, the feature vector will be a vector of length equals to 50. Each feature element will be represented as the distance from the orientation point to one of the fifty points on the contour as shown in figure 10.

For the second category, with the wrist appears at from bottom-right, the case will be similar to the previous category. However, due to that the orientation point is at the most right column of the last row in the contour image and the total angle (Big_{theta}) is equal to 90 degrees, the angle in this case ranges from zero to 90 degrees as shown in figure 11. The feature vector length is also equal to 50 and calculated by the same way as the previous category.

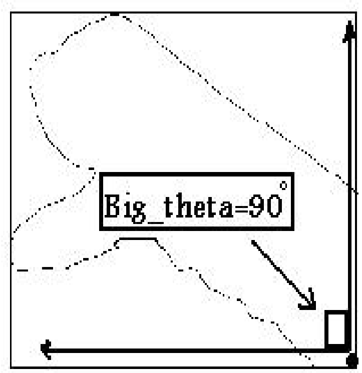


Figure 11. The second category: wrist appears from bottom-right and Big_{theta} =90°

For the third category, with the wrist appears at from bottom-left, the case will be similar to the second category. However, due to that the orientation point is at the most left column of the last row in the contour image and the total angle (Big_{theta}) is equal to 90 degrees, the minimum angle is equal 90 degrees and the maximum angle in this case is equal to 180 degrees as shown in figure 12.

For making the extracted features scaling invariant, the system selects the maximum value in the each feature vector and divide all the elements of that feature vector by the selected maximum value.

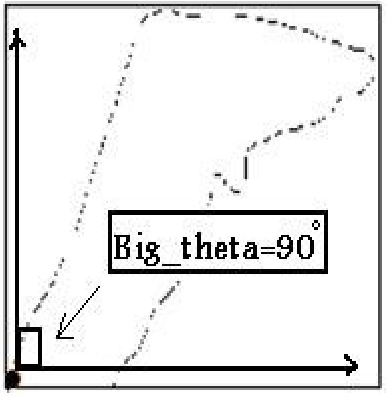


Figure 12. The third category: wrist appears from bottom-left and Big_{theta} =90°

E. Classification phase

Two different classifiers namely minimum distance classifier (MDC) and multilayer perceptron (MLP) neural network have been used. MDC is a traditional nonparametric statistical classifier. MLP is a well-known neural network classifier.

1) Minimum Distance Classifier

The minimum distance classifier (MDC) is an example of a commonly used 'conventional' classifier [20], [21]. The single nearest neighbor technique completely bypass the problem of probability distance and simply classifies any unknown sample as belonging to the same class of the most similar or nearest feature vector in the training set of data [22]. Nearest can be taken to the smallest Euclidean distance in n-dimensional feature space and the classifier compares the detected feature vector X with all the class known feature vectors y_i, and minimizes the discriminant of minimum distance classifier using equation (5).

$$Distance = \sqrt{\sum_{i=0}^{N}(X_i - y_i)^2} \qquad (5)$$

Where N is the feature vector length that is equal 50, $1 \leq i \leq N$. The minimum value is used in conjunction with a lookup letter table to select the appropriate letter to classify.

2) Neural Network

The Artificial Neural Network or ANN algorithms are the commonly used as base classifiers in classification problems [23]. An artificial neural network is a powerful data modeling and information-processing paradigm that is able to capture and represent complex input/output relationships [24]. The advantage of neural networks mainly lies in that they are data driven self-adaptive methods, which can adjust themselves to the data without any explicit specification of functional or distributional form for the underlying model. Also, they are universal functional approximators in that neural networks can approximate any function with arbitrary accuracy [24], [25]. The function of the neural network is transforming inputs into meaningful outputs. It inspired by the way of biological nervous systems, neural networks look like the human brain in two stages learning stage and testing stage.

Moreover, neural network is able to represent both linear and non-linear relationships and the way it can learn these relationships directly from the modeled data. The most common neural network model is the multilayer perceptron (MLP) shown in figure 13. This type called supervised network because it requires a desired output in order to learn. The goal of this type of network is to create a model that correctly maps the feature vector of a single sample (input) to the class of the input sample (output) using historical data so that the model can then be used to produce the output when the desired output is unknown.

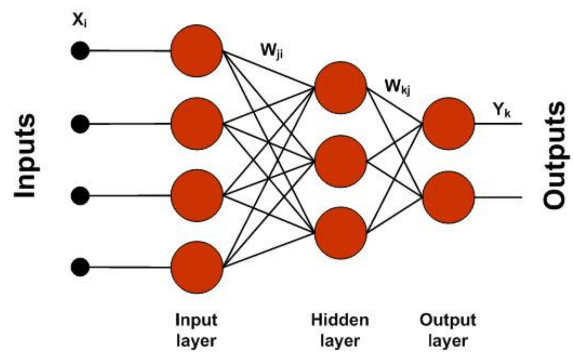


Figure 13. Feed-forward multilayer perceptron (MLP)

The neural network consists of three types of layers input, hidden and output layer. Each of them consists of number of perceptrons or neurons and these neurons connected together from layer according to specific network architecture as in figure 13. Each connection has a very important unit called "weight". The weight unit controls the degree of intelligence of the neural network.

The input layer is the layer that represents the input data so the length of this layer is equal to the length of the input data (feature vector), and there is only one input layer in the neural network. It consists of a set of input values (X_i) and associated weights (W_i).

The hidden layer is the kernel of the network because it controls the number of thinking equations and by which the result gets better. The neural network may contain several hidden layers. The last one is the output layer, which returns the result. The length of this layer equals to the number of classes. There is only one output layer in the neural network.

The MLP neural network looks like any neural network so it goes through two stages. The first one is the learning stage, which trains the network to be able to think and return the best result. The learning process comes by updating the weights.

IV. Experimental Results

The proposed system was implemented on a Penitum4 (2.6GHz) desktop computer with Microsoft Windows XP (SP2) platform using MathWorks MATLAB v.7.10 (R2010a) and Java JDK v. 1.6.

To evaluate the performance of the proposed system, several videos containing sequences of letters such as "Ra, Lam, Kaf" have been classified. The alphabet used for experiments has been constrained, for simplicity, to a training set of 15 letters and the system runs using these letters. Due to the overlap of training letters in the feature space, each video has been classified again using the constrained training set containing only the letters used in the word. For example, for the letters “Ra, Lam, Kaf” the system detected the “Ra, Lam and Kaf” classes as shown in figure 14.

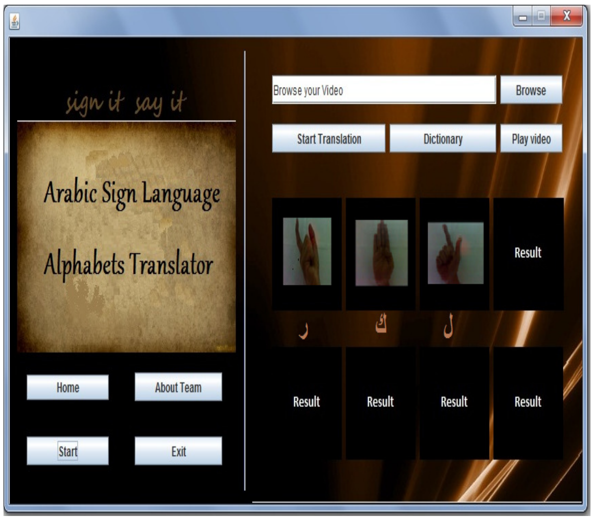


Figure 14. Detection and translation of the “Ra, Lam, and Kaf” letters

Table I demonstrates performance measures of the ArSLAT system. It is obvious that sign translation accuracy using MDC (91.3%) is higher than the accuracy achieved using MLP (83.7%), as MDC matches each unknown letter with all the known letters in the same category in the database and takes the nearest one to this letter, however it takes longer time than MLP.

Table 1. Retrieved results, where t_l is the total number of detected letters, t_c is the total correct letters, and t_f is the total false letters

Video duration	Classifier	T_L	T_C	T_F	Accuracy
30 min.	MDC	1000	913	87	91.3 %
30 min.	MLP	1000	837	163	83.7%

V. Conclusions and Future Work

In this paper, a system for the purpose of the recognition and translation of the alphabets in the Arabic sign language were designed. The proposed Arabic Sign Language Alphabets Translator (ArSLAT) system is composed of five main phases; Pre-processing phase, Best-frame Detection phase, Category Detection phase, Feature Extraction phase, and finally Classification phase. The extracted features are translation, scale, and rotation invariant, which make the system more flexible. Experiments revealed that the proposed ArSLAT system was able to recognize a representing subset (15 letters) of the Arabic manual alphabets with an accuracy of 91.3% and 83.7% using minimum distance classifier (MDC) and multilayer perceptron (MLP), respectively.

There still a lot of room for further research in performance improvement considering different feature sets and classifiers. Moreover, additional improvements can be applied for this system to be used for mobile applications to provide easy communication way among deaf/hearing-impaired people. Also, this system could be developed to be provided as a web service used in the field of conferences and meetings attended by deaf people. Furthermore, this system could be used by deaf and normal people for controlling their computers and performing actions to them without the need for touching any device. Finally, it can be used in intelligent classrooms and intelligent environments for real-time translation for sign language.

References

[1] K. Assaleh, and M. Al-Rousan. "Recognition of Arabic Sign Language Alphabet Using Polynomial Classifiers", *EURASIP Journal on Applied Signal Processing (JASP)*, 2005(13), pp. 2136-2145, 2005.

[2] M. AL-Rousan, K. Assaleh, and A. Tala’a. "Video-based Signer-independent Arabic Sign Language Recognition Using Hidden Markov Models", *Applied Soft Computing*, 9(3), pp. 990-999, 2009.

[3] O. Al-Jarrah and A. Halawani. "Recognition of Gestures in Arabic Sign Language Using Neuro-Fuzzy Systems", *Artificial Intelligence*, 133(1-2), pp. 117-138, 2001.

[4] M. Vermeerbergen. "Past and Current Trends in Sign Language Research”, *Language & Communication*, 26(2), pp. 168-192, 2006.

[5] J. Davis and M. Shah. "Visual Gesture Recognition", *IEE Proceedings: Vision, Image and Signal Processing*, 141(2), pp. 101-106, 1994.

[6] Y. Wu and T. S. Huang. "Vision-based Gesture Recognition: A Review", In *Proceedings of 3rd International Gesture Workshop (GW'99)*, pp. 103-115, France, March 1999.

[7] S. M. Halawani. "Arabic Sign Language Translation System on Mobile Devices", *International Journal of Computer Science and Network Security (IJCSNS)*, 8(1), pp. 251-256, 2008.

[8] V. I. Pavlovic, R. Sharma, and T. S. Huang. "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review", *IEEE Trans. Pattern Anal. Machine Intell.*, 19(7), pp. 677-695, 1997.

[9] S. S. Fels and G. E. Hinton. "Glove-talk: A Neural Network Interface between a Data-glove and a Speech Synthesizer", *IEEE Trans. Neural Networks*, 4(1), pp. 2-8, 1993.

[10] D. J. Sturman and D. Zeltzer. "A Survey of Glove-based Input", *IEEE Comput. Graph. Appl.*, 14(1), pp. 30-39, 1994.

[11] D. L. Quam. "Gesture Recognition with a Dataglove", In *Proceedings of IEEE National Aerospace and Electronics Conference (NAECON '90)*, vol. 2, pp. 755-760, Dayton, Ohio, USA, May 1990.

[12] J. Eisenstein, S. Ghandeharizadeh, L. Huang, C. Shahabi, G. Shanbhag, and R. Zimmermann. "Analysis of Clustering Techniques to Detect Hand Signs", In *Proceedings of International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP'01)*, pp. 259-262, Hong Kong, China, May 2001.

[13] M. Huenerfauth. "Generating American Sign Language Classifier Predicates For English-To ASL Machine Translation", Ph.D dissertation, University of Pennsylvania, Department of Computer and Information Science, Philadelphia, PA, USA, 2006.

[14] J.A. Bangham, S.J. Cox, M. Lincoln, I. Marshall, M. Tutt, and M. Wells. "Signing for the Deaf Using Virtual Humans", *IEE Seminar on Speech and Language Processing for Disabled and Elderly People*, no. 2000/025, London, UK, pp. 4/1-4/5, April, 2000.

[15] S. C.W. Ong and Surendra Ranganath. "Automatic Sign Language Analysis: A Survey and the Future beyond Lexical Meaning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6), June, 2005.

[16] S. Phitakwinai, S. Auephanwiriyakul, and N. Theera-Umpon. "Thai Sign Language Translation Using Fuzzy C-Means and Scale Invariant Feature Transform", In *Proceedings of International Conference of Computational Science and Its Applications - ICCSA 2008*, Lecture Notes in Computer Science, vol. 5073/2008, pp. 1107-1119, 2008.

[17] V. S. Kulkarni and S.D. Lokhande. "Appearance Based Recognition of American Sign Language Using Gesture Segmentation", *International Journal on Computer Science and Engineering (IJCSE)*, 2(3), pp. 560-565, 2010.

[18] S. Morrissey and A. Way. "Joining Hands: Developing a Sign Language Machine Translation System with and for the Deaf Community", In *Proceedings of Conference and Workshop on Assistive Technologies for People with Vision and Hearing Impairments: Assistive Technology for All Ages (CVHI-2007)*, Granada, Spain, pp. 28-31, August, 2007.

[19] N. El-Bendary, A. E. Hassanien, H. Zawbaa, M. Daoud, and K. Nakamatsu. "ArSLAT: Arabic Sign Language Alphabets Translator," In *Proceedings of The International Conference on Computer Information Systems and Industrial Management Applications (CISIM)*, Krackow, Poland, pp. 590 - 595, 8-10 October, 2010.

[20] M. S. Packianather and P. R. Drake. "Comparison of Neural and Minimum Distance Classifiers in Wood Veneer Defect Identification", The Institution of Mechanical Engineers, Part B: *Journal of Engineering Manufacture*, Sage Publications, 219(11), pp. 831-841, 2005.

[21] R. Boveiri. "Persian Printed Numerals Classification Using Extended Moment Invariants", In *Proceedings of WASET Int. Conf. on Image and Vision Computing*, Rio de Janeiro, pp. 167-74, 2010.

[22] H. Parvin, H. Alizadeh and B. Minaei-Bidgoli. "A New Divide and Conquer Based Classification for OCR", In *Proceedings of Convergence and Hybrid Information Technologies INTECH*, Croatia, pp. 426, March, 2010.

[23] F. Roli, G. Giacinto, and G. Vernazza. "Methods for Designing Multiple Classifier Systems", In *Proceedings of 2nd International Workshop on Multiple Classifier Systems*, Lecture Notes in Computer Science, Cambridge, UK, Springer-Verlag, vol. 2096, pp. 78–87, 2001.

[24] G. P. Zhang, "Neural Networks for Classification: A Survey", *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications And Reviews*, 30(4), pp. 451-462, November, 2000.

[25] K. Hornik. "Approximation Capabilities of Multilayer Feedforward Networks", *Neural Networks*, vol. 4, pp. 251–257, 1991.

Author Biographies

Dr. Nashwa El-Bendary She was born in Cairo, Egypt, December 1979. She received her M.Sc. degree in 2004 and Ph.D. degree in 2008, both in Information Technology from the Faculty of Computers and Information, Cairo University, Egypt. Currently, she is an assistant professor at the Arab Academy for Science, Technology, and Maritime Transport (AASTMT), Cairo, Egypt. She has published several papers in major international journals and peer-reviewed international conference proceedings. Her main research interests are in the areas of biometrics, intelligent environments, information security, and wireless sensor networks. Dr. Nashwa is a member of the editorial boards of a number of international journals. She has also been a reviewer, program committee member, and special session co-chair in several international conferences.

Hossam M. Zawbaa He is a Master's student at Faculty of Computers and Information, Cairo University, Egypt. His advisor is prof. Aboul Ella Hassanien and his current research interests are data and text mining, video and image processing, and optical character recognition. He graduated in 2008 from the Faculty of Computers and Information, Cairo University, Egypt. He has published some papers in a number of international journals and peer-reviewed international conference proceedings.

Mahmoud S. Daoud He is a final year undergraduate student of Information Technology Dept., Faculty of Computers and Information, Cairo University, Egypt. His current research interests are image processing and pattern recognition.

Prof. Aboul Ella Hassanien He received his B.Sc. with honours in 1986 and M.Sc degree in 1993, both from Faculty of Science, Ain Sham University, Egypt. On September 1998, he received his doctoral degree from the Department of Computer Science, Graduate School of Science & Engineering, Tokyo Institute of Technology, Japan. He has authored/coauthored over 160 research publications in peer-reviewed reputed journals and conference proceedings. He serves on the editorial board and reviewer of number of journals and on the program committee of several international conferences and he has editing/written more than 18 books. He has received the excellence younger researcher award from Kuwait University. Also, he has guest edited several special issues on various topics. His research interests include, rough set theory, wavelet theory, medical image analysis, intelligent environment, multimedia data mining, and cyber security.

Prof. Kazumi Nakamatsu He was born in Shizuoka, Japan, December 1953. He has earned his BA in Informatics (Shizuoka Univ., Japan, 1976), MS in Informatics (Shizuoka Univ., Japan, 1978), and PhD in Computer Science (Kyushu Univ., Japan, 1999). He has studied application of logic, especially application of his own paraconsistent annotated logic program called EVALPSN, and applied it to various kinds of intelligent information systems.