

A Proposed Model for Standard Arabic Sign Language Recognition Based on Multiplicative Neural Network

A. S. Elons^{*1}, Magdy Aboul-ela^{**2}, M. F. Tolba^{*3}

** Scientific Computing Department- Faculty of Computers and Information Sciences- Ain Shams University-Cairo-Egypt*

¹ahmed.new80@hotmail.com

³fahmytolba@gmail.com

*** Sadat Academy-Cairo-Egypt*

²maboulela@gmail.com

Abstract: Sign language recognition is one of the most challenging fields in Human-Computer Interface (HCI) applications. Although there are many obstacles that could dramatically limit the spread of sign language translators in our daily life, the community needs for these translators are no longer a luxury and increase day after day, other than the problems of sign languages all over the world, Arabic sign language enjoys its own difficulties and issues. This paper discusses Arabic sign language problems and proposes a recognition model for standard Arabic sign language. A model is proposed and developed for real-time hand signs recognition. The experiment was conducted on 100 signs and the result was 94% recognition accuracy confirming words offline extensibility. Although the scientific understanding for the sign language is an essential step to build up a realistic recognition system, the proposed model can be used in other sign languages. The model exploits multi-stage Multiplicative Neural Networks for posture classification.

Key words: Arabic Sign Language (ArSL), Multiplicative Neural Network (MNN), Graph Matching, Posture, Gesture.

1 INTRODUCTION

Currently, there are 150 million deaf people around the world, including three millions in Egypt [1]. In response to the increasing number of deaf people in Egypt, sign language interpreters should be made readily available. Sign language is a kind of gesture and is one of the most natural means of exchanging information for most deaf people. The aim of sign language recognition is to provide an efficient and accurate mechanism to transcribe sign language into text or speech. Sign language is a visual and manual language, made up of signs created with the hands, facial expressions, body posture and movements [1, 2]. Sign language conveys ideas, information and emotion with as much range, complexity and versatility as spoken languages [2]. Previous trials for sign language interpretation could not gain industry satisfaction to put these trials into a product form. N. Tanibata et al. [3] in 2001, proposed a method of extraction for hand features and recognition of JSL (Japanese Sign Language) words. For tracking the face and hand, they could recognize 64 out of 65 words successfully by 98.4%. Chen et al [4] introduced in 2003 a system for recognizing dynamic gestures (word signs) for TSL (Taiwanese Sign Language). They used frequency domain features (Fourier Transform) plus some information from motion analysis for recognizing 20 words. An average of 92.5% recognition rate was achieved. In 2004 and 2005, J. Zieren et al. [5, 6] presented two systems for isolated recognition; the first is for recognizing GSL on a vocabulary of 152 signs and achieved a rate of 97% using HMM but the rate decreases for the group of signs that contain overlaps in either hands or face and hands.

When Microsoft released "Kinect" in November 2010 [7], it mainly targeted consumers owning a Microsoft Xbox 360 console, allowing the user to interact with the system using gestures and speech [7]. The device itself features an RGB camera, a depth sensor and a multi-array microphone and is capable of tracking the user's body movements. This new technology encouraged researchers in sign language recognition to customize it in real time recognition. Natheer [8] at el Jordan University of Science and Technology developed a Kinect demo for Arabic Sign Language recognition that is capable of tracking user's fingers. However, the application used a very limited dataset. Unfortunately, due to the low resolution of Kinect cameras, hand details and shapes in fast time animation are indistinguishable. Here is an example of images that are captured using a Kinect camera in Fig. 1, the finger locations and shapes in the images can't be determined.



Figure 1: Kinect images

In sign language recognition, the real time response of the system for the interaction task is most important providing an immediate feedback to the user. Consequently, tracking and recognition must be handled as fast as possible, preferably at the same rate as images are displayed or obtained. Otherwise, the user could be easily confused or irritated and would not know whether he or she has already initiated an action. As a result, it is important to handle all necessary calculations within a given time frame. Although modern CPUs are becoming increasingly faster, it is sometimes difficult to process all incoming and outgoing data in real time. Many researches now explore the potentials of the current graphic processing units (GPUs). Furthermore, by now the GPU is not only utilized for graphical purposes but also for performing highly parallel tasks efficiently. The usage of graphic cards for such a purpose is defined as general purpose computation on the GPU. Textures or in this case, image data obtained from digital cameras or webcams, are well suited for the computation and execution of multiple operations in parallel on the GPU. Park et al. [9] implemented a real time embedded FPGA based gesture recognition system using a 5DT data glove. This approach is used in order to reduce the problems of space, movement and lighting limitations. The architecture of the system consists of three main modules that are; input, recognition and display modules. The system recognizes the hand gesture by performing data calculations with a checksum function on the input data and compares the result to the header byte before it proceeds to the matching process. The matching process compares the input hand gesture with the pre-defined hand gesture, then, the result is displayed on the LCD screen. Yu et al. [10-12] introduced an FPGA based smart camera which is called the "GestureCam". This smart camera is capable of performing simple vision based hand gesture recognitions. GestureCam comprises an image capture unit (ICU), FPGA based gesture recognition unit, (GRU) and a host and display unit (HDU). A low pass filter is applied in order to obtain a clearer skin image from the extracted skin color image. GestureCam applies a contour tracing algorithm, which is based on an inner boundary tracing algorithm (IBTA) with several modifications, to extract hand contours for gesture classification. The GestureCam classification module is using ANN and trajectory based methods.

2 STANDARD ARABIC SIGN LANGUAGE

Luckily, Sign language in the Arab World has been recently recognized and documented. Many efforts have been made to establish the sign language used in individual countries, including Jordan, Egypt, Libya 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. In 2005 the "League of Arab States" accredited and officially published a unified Arabic Sign Language dictionary that contained 1000 words. A second edition was released in 2008 [2].

There are some issues that differentiate Arabic sign language from other sign languages:

- Arabic Sign Language was not so well defined until recently.
- In Arabic Sign Language, the two hands can be used interchangeably to express the same gesture. This property makes two gestures representing the same meaning, Fig 2.

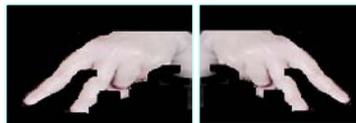


Figure2: Using 2 hands for the same sign

Arabic Sign language is a "descriptive" language and as such, the gestures try to describe the meaning, for instance Fig 3. This property makes the gesture too long.



Figure3: A signer that describes a "watch"

- Facial expressions or the human body play an important role in sign understanding.

Compared to other sign languages, not much has been done in automating Arabic sign language, except for a few individual attempts. M. Al-Rousan et al. [13] and O. Aljarrah et al. [14] in 2000 and 2001 respectively developed two systems for recognizing 30 static gestures of Arabic sign language using a collection of Adaptive Neuro-Fuzzy Inference System (ANFIS) networks for training and classification depending on the spatial domain features. Tolba et al. [15] used PCNN for the recognition of 30 alphabet postures and gained 93% recognition accuracy.

3 RECOGNITION SYSTEM

Many (SL) recognition systems have implemented techniques to track and recognize hand signs. In this paper, the proposed recognition system considers the different environment capturing factors; lighting conditions, distance, translation, rotation and non-uniform backgrounds. Figure 4 illustrates a recognition system diagram.

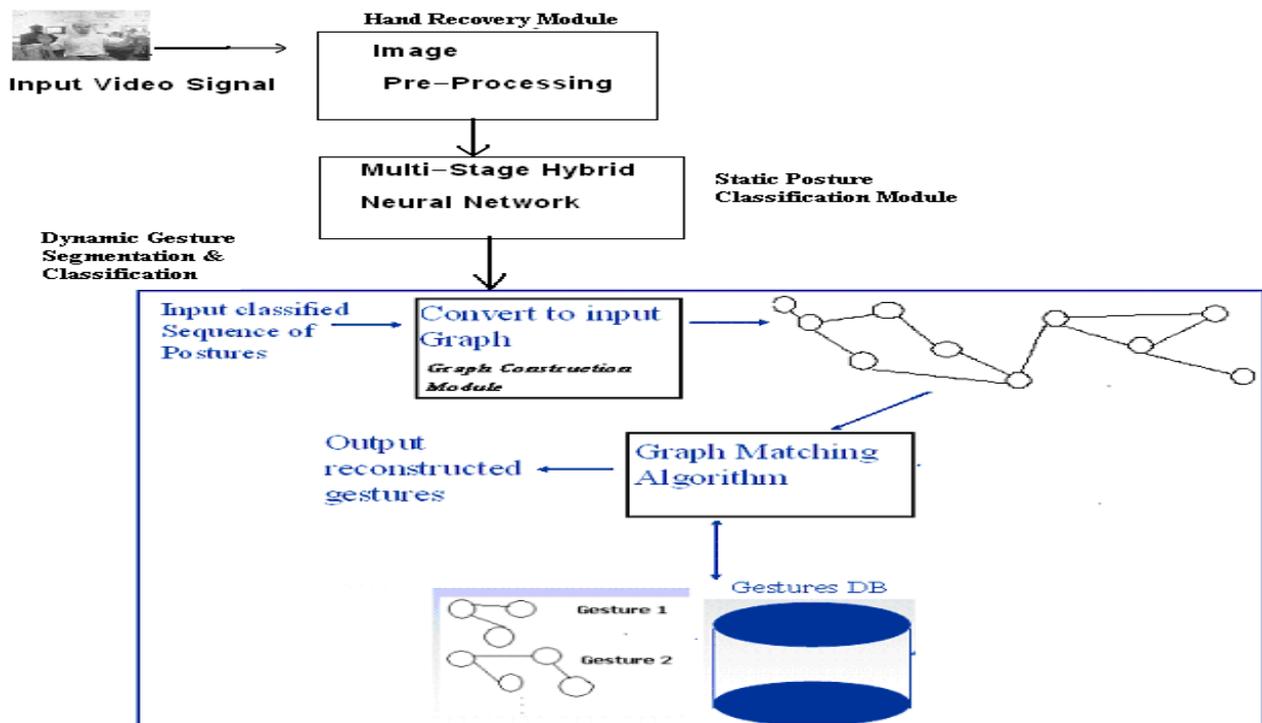


Figure 4: The recognition system

A. PCNN

A pulse coupled neural network (PCNN) is a model of a biological network, specifically a model of a fragment of the cat's sight network. It is a single layer network [16] composed of neurons. Each of them is linked to one pixel of the input image. Each neuron contains two input compartments; the feeding and the linking. The feeding receives an external stimulus as well as local stimulus while the linking only receives local stimulus [17]. The local stimulus comes from the neurons within a feeding radius. This local stimulus is hereafter called the firing information. The external stimulus is the intensity from the corresponding pixels in the picture. The feeding and linking are combined in a second order fashion to create the potential which then decides together with the output, on whether the neuron should fire or not. There are several differences between the algorithms for the modified PCNN neuron and the exact physiological pulse coupled neuron. The differences are due to several simplifications made to the calculations while still keeping the main features of the general theory. Each neuron in the modified PCNN could be described by the following set of equations [14]:

$$L(i) = L(i-1) \cdot e^{-\alpha L} + VL \cdot (R * Y_{sur}(i-1)) \quad (1)$$

$$F(i) = S + F(i-1) \cdot e^{-\alpha F} + VF \cdot (R * Y_{sur}(i-1)) \quad (2)$$

$$U(i) = F(i) \cdot [1 + \beta \cdot L(i)] \quad (3)$$

$$\theta(i) = \theta(i-1) \cdot e^{-\alpha \theta} + V\theta \cdot Y_{out}(i-1) \quad (4)$$

$$U > \theta(i) \Rightarrow Y_{out} = 1 \quad (\text{Firing Condition}) \quad \text{otherwise} \Rightarrow Y_{out} = 0 \quad (5)$$

where $L(i)$ is the input linking potential, $F(i)$ is the input feeding potential and S represents the intensity of the given image element. $U(i)$ is the activation potential of a neuron, $\theta(i)$ is the threshold potential of a neuron and (i) is the iteration step. Parameters (αL) , (αF) and $(\alpha \theta)$ are decay coefficients, (β) is linking the coefficient and parameters (VL) and (VF) are coefficients of the linking and threshold potential. Y_{sur} is the firing information that indicates whether the surrounding neurons have fired or not and (Y_{out}) indicates whether this neuron fires or not. (R) is the matrix of weight coefficients and $*$ is the convolution operator. An example of the modified PCNN neuron architecture is shown in Fig 5 as a schematic block diagram of the modified PCNN neuron as described through (1) - (5).

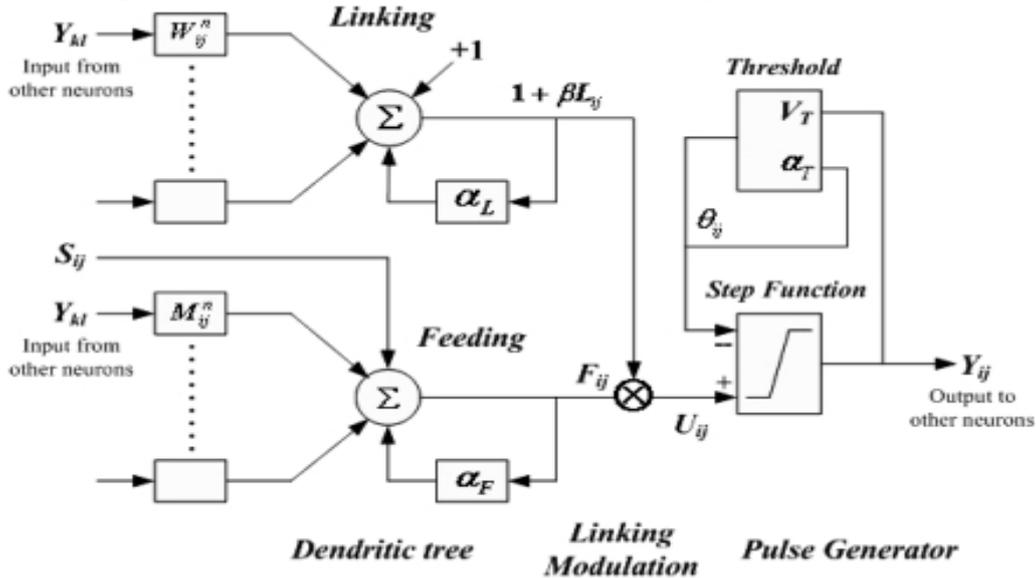


Figure 5: PCNN Neuron Model

B. PCNN and Feature Generation

PCNN generates a unique one-dimensional time series (image signature) based on computing the global pulsing behavior. The idea of image signatures stems from biological research done by McClurken et al. [15]. The creation of signatures and using them in object recognition with the PCNN was first proposed by Johnson [18]. The standard approach of feature generation $G(n)$ of specific iteration (i) is based on a series of virtual binary image generation. It is calculated as the sum of output quantities (Y_i) of activated neurons in the given iteration step [19, 20]:

$$G(n) = \sum_{i=0}^n Y(i) \quad (6)$$

Through influence of geometrical transforms, it is very important for the standardization of the generated features by a standard equation.

$$g(n) = \frac{G(n)}{\max(G)} \quad (7)$$

where $\max(G)$ is the function that returns the maximal value in the feature space for the first impulse of the function $G(n)$. A modification can be applied to (7), such that:

$$g(n) = \frac{G(n)}{\sum_{ij} S_{ij}} \quad (8)$$

where (S_{ij}) is the intensity of a given image pixel (i,j) ; the previous equation achieves the standardization but it does not satisfy the condition of $0 \leq g(n) \leq 1$. $0 \leq g(n)$ but $g(n) \leq 1$ is also not valid in all the cases. Froge [15] proposed a new form of feature generation by introducing a new equation for feature value calculation $g(n)$:

$$g(n) = \frac{\sum_{i=1}^n X(i) \times Y(i)}{\sum_{ij} S_{ij}} \tag{9}$$

Besides, Froge checked the validity of the standardization condition $0.5 < (X(i) \times Y(i)) < 1$ and he showed the invariance against non-standard cases. Tolba et al [2] added the "continuity factor" to Froge's equation.

$$g(n) = \frac{\sum_{i=1}^n (X(i) \times Y(i) \times CF(i))}{\sum_{ij} S_{ij}} \tag{10}$$

where CF(i) is the continuity factor at iteration (i) and it represents the weight for pulses in iteration (i). Tolba et al [2] showed that eq(10) neutralizes the lighting condition and background effect and minimizes the "interference" effect between background pixels and object pixels. In this paper, eq (10) will be used in the feature extraction step using PCNN.

C. Multi-Stage Multiplicative Neural Network (MMNN)

The McCulloch-Pitts model initiated the use of weighted summing units as the basic neuron model [20]. It is commonly known that there is only a minor closeness between this neuron model and the behavior of real biological neurons. In particular, the interaction of synaptic inputs is known to be non-linear. In search of biologically closer models of neural interactions, neurobiologists have found that multiplicative operations play an important role in single neuron computations. It mainly reflects how complex behavior can emerge in simple networks. Recently, evidence has accumulated regarding specific neurons in the nervous system of several animals computed in a multiplicative manner [21]. Multiplication increases the computational power and storage capacity of neural networks. The basic building block of a single neuron MNN is depicted in Fig 6.

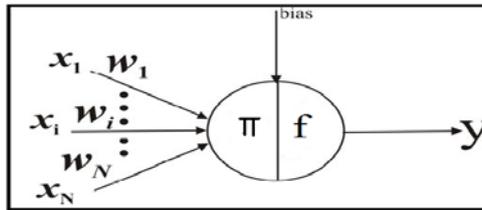


Figure 6: MNN Neuron Model

A node receives a number of real inputs X_1, X_2, \dots, X_N which are then multiplied by a set of weights W_1, W_2, \dots, W_N and bias terms b_1, b_2, \dots, b_N are then added. The back propagation learning algorithm with multiplicative neural networks (MNN) has to be modified to be more efficient. The resultant values are multiplied to form a polynomial structure. This output of the node is further subjected to the non-linear function f defined as

$$f = \frac{1 - e^{-x}}{1 + e^{-x}} \tag{11}$$

In the MNN, the numbers of nodes described above are arranged in layers. A vector input is passed to each node of the first layer. The outputs of the first layer nodes then become inputs to the nodes in the following layer and so on. Weighted connections exist from a node to every node in the succeeding node but no connections exist between nodes of the same layer. As the number of layers in the MNN increases, decision regions are formed which are considerably more complex and have highly non-linear boundaries. Fig. 7 shows a general model of a feed forward MNN.

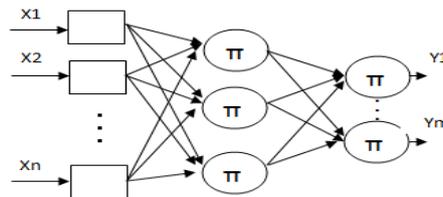


Figure 7: Architecture of feed forward MNN

The MNN is trained using a supervised learning algorithm where a set of inputs and the corresponding target vectors are used to adjust the scalar parameters weight and bias. The network is trained incrementally so that the weights and biases are modified after each input is presented. As the inputs are supported to the network, the network actual outputs are

compared to the desired targets. The learning rule is then used to adjust the weights and biases of the network in order to move the network outputs closer to the targets.

The output of the node u before applying the activation function is given by [21]

$$U = \prod_{i=1}^n (w_i x_i + b_i) \quad (12)$$

The bipolar sigmoid activation function f is given by:

$$f(u) = \frac{1 - e^{-u}}{1 + e^{-u}} \quad (13)$$

An error back propagation based learning rule is used for training. The MSE is given by [20]:

$$E = \frac{1}{2M} \sum_{i=1}^M (y_i - d_i)^2 \quad (14)$$

where M is the number of input patterns. The weight update equation for a single layer algorithm is given by:

$$\Delta W = -\eta \frac{dE}{dW_i} \quad (15)$$

where, η is the learning rate: If η has a large value, learning occurs quickly but if it has a much larger value it may lead to instability and errors may even increase. The bias is updated as:

$$\Delta b = -\eta \frac{dE}{db_i} \quad (16)$$

The new weight and bias is calculated by:

$$W_{new} = W_{old} + \Delta W \quad (17)$$

$$b_{new} = b_{old} + \Delta b \quad (18)$$

The word Multi-stage corresponds to Multi-level, in the case of Multi-stage Neural Networks there are two or more levels of networks, each level (stage) gives more information about the solution until the solution is obtained by reaching the last level (stage). The Multi-stage Neural Networks classifier contains multiple levels, the first level acts as a selector, the result of the selector represents an enabling signal for the rest of the modular networks and the input pattern propagates through the next levels in an incremental way until the elementary network is reached. The approach of decomposing the problem and applying a modular structure of network has a number of benefits.

- 1) As demonstrated on the task of speech recognition by Waibeletal (1989), it allows the use of several smaller networks rather than one massive network and thereby reduces the amount of training time and data acquired, it may also result in a superior classification accuracy.
- 2) It produces a system which can more easily be extended. Each network can be retrained, without retraining the networks in the same level.

A Multi-stage Neural Networks Classifier structure has obtained very good results within the speech recognition community. Due to the parallelism that exists between the speech community and vision community; it can be used in different vision systems, for instance; Optical character recognition, face recognition and hand gesture recognition. Due to the large size of (ASL) postures, postures can be clustered according to different criteria's. In the proposed system multi-structures have been applied and tested, they differ in the way of clustering the postures. In the proposed system postures are clustered into 2 clusters; a cluster for one hand postures and the other for both hands postures, Fig 8 and use 2 level multi-stage networks, the first level contains a selector that decides whether the input pattern is a posture of one or two hands, the second level contains two networks, one network to classify classes of a one hand cluster and the other one to classify classes of two hand clusters. This criterion of clustering was used in the proposed system. The implementation details of each network are described in the experimental results section. Also, a sensitivity analysis for neural network parameters is performed to guarantee optimal recognition accuracy.



Figure 8: Clustering based on number of hands

D. Sign Language Recognition and Graph Matching

Graphs are a general and powerful data structure useful for the representation of various objects and concepts in pattern recognition and machine vision. In a general representation, the nodes typically represent objects or parts of objects, while the edges describe relations between objects or object parts. Graphs have some interesting invariance properties. For example, if a graph which is drawn on paper and translated, rotated or transformed into a mirror image, it's still the same graph in the mathematical sense. These invariance properties as well as the fact that graphs are very well suited to model complex objects in terms of parts and their relations, make them very attractive for various applications. Graph representation and graph matching have been successfully applied to a large number of problems in computer vision and pattern recognition. Examples include character and schematic diagram interpretation, shape analysis, image registration and 3-D object recognition.

Let: $G = (V, E, \mu, \nu, L_n, L_e)$ be a model graph and M it's corresponding $n \times n$ - adjacency matrix. Furthermore, let $A(G)$ denote the set of all permuted adjacency, matrices of G , $A(G) = \{M_P | M_P = PMP^T \text{ where } P \text{ is a } n \times n \text{ permutation matrix}\}$. The total number of permuted adjacency matrices is $|A(G)| = n!$ as there are $n!$ different permutation matrices of dimension n . For the model graph G with corresponding $n \times n$ adjacency matrix M and the input graph G_I with an $m \times m$ adjacency matrix M_I and $m \leq n$, determines whether there exists a matrix $M_P \in A(G)$ such that $M_I = S_{m,m}(M_P)$. If such a matrix M_P exists, the permutation matrix P corresponding to M_P describes a sub-graph isomorphism from G_I to G , i.e. $M_I = S_{m,m}(M_P) = S_{m,m}(PMP^T)$. If G_I and G are of equal size, the permutation matrix P represents a graph isomorphism between G_I and G , i.e. $M_I = PMP^T$. We propose to organize the set $A(G)$ in a decision tree that each matrix in $A(G)$ is classified by the tree. The features that will be used for the classification process are the individual elements in the adjacency matrices, we introduce a new notation for an $n \times n$ adjacency matrix $M = (m_{ij})$. We say that the matrix consists of an array of so called **row-column** elements a_i , where each a_i is a vector of the form:

$$a_i = (m_{1i}, m_{2i}, \dots, m_{ii}, m_{i(i-1)}, \dots, m_{i1})$$

The matrix can then be written as: $M = (a_1, a_2, \dots, a_n)$; $i = 1, \dots, n$ [22, 23]. Figure 9 illustrates the structure of an adjacency matrix M with regard to its row-column elements [24].

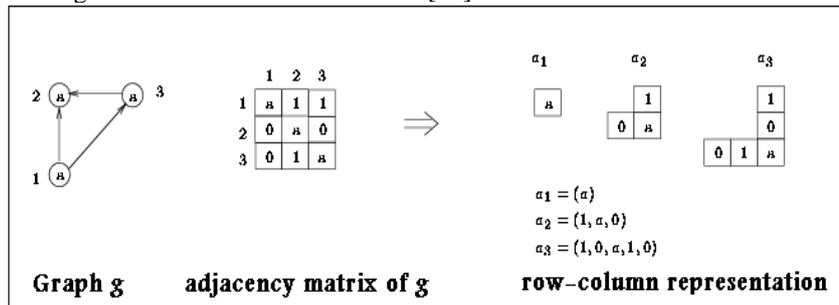


Figure 9: The row-column representation of the adjacency matrix

4 EXPERIMENTAL RESULTS

The first study is conducted to collect ASL static postures which form most of ASL words, 135 clips were captured containing a total of 300 hand signs. This study concludes that most gestures are a combination of 158 postures, 88 single hand postures (44 left hand gestures, 44 right hand gestures) and 70 postures use both hands. 250 gestures were captured for this study. The implementation details of the PCNN for the features generation step are: $\beta=0.1, \alpha_L=1.0, \alpha_q=0.8, V_L=0.25, V_F=0.6$ and $V_\theta=20$. The MNN has been trained and compared with traditional Multi-layer perceptron networks (MLP) with respect to both training time and recognition rate. Table 1 illustrates the training times for both MNN and MLP against different dataset sizes.

TABLE 1
TRAINING TIME for MNN and MLP

Dataset size	MNN time (hour: min)	MLP Time (hour: min)
10	1:42	1:01
40	3:55	2:59
70	6:21	4:01
100	9:02	5:29
120	10:58	6:04
140	12:34	7:08
158	15:22	8:12

The training time has been computed with the number of selected features which are 11. Table 2 illustrates the recognition accuracy of MNN for different feature numbers (5, 8, 11, 18 and 25).

TABLE 2
RECOGNITION ACCURACY

No of Nodes	No of Epochs	5 features	8 features	11 features	18 features	25 features
Accuracy %						
5	20	72	74	88	87	86
	40	75	75	88	84	86
	60	75	80	89	88	87
	80	75	83	90	88	86
	100	75	83	90	86	86
10	20	78	85	90	89	82
	40	78	86	90	89	82
	60	76	86	91	88	81
	80	79	86	91	87	82.
	100	79	86	92	84	80
15	20	82	87	92	87	85
	40	82	84	93	87	81
	60	82	84	93	87	81
	80	83	84	93	87	84
	100	83	84	94	89	84
20	20	83	86	92	89	79
	40	83	86	93	89	76
	60	82	86	93	89	75
	80	83	86	93	90	79
	100	83	84	93	90	81
25	20	82	86	93	90	79
	40	80	86	92	89	77
	60	79	88	89	89	77
	80	80	86	89	90	77
	100	81	88	89	90	77
30	20	84	87	89	88	78
	20	84	87	91	88	76
	40	83	87	91	87	78
	60	83	87	91	87	77
	80	83	87	91	87	78
100	83	87	92	87	77	

The comparison between the 11 features set and 25 features set shows that an increase in the number of features will not necessarily increase the classification accuracy for the same number of hidden nodes. In addition, a larger features set will increase the processing time while results are similar. Thus, in the current research, the 11 features set is selected as the optimized set of input features to train the network and classify the postures. Based on the previous table, the implementation details of the MNN network are: 1 hidden layer has been used and the Sigmoid Activation function was used. The network has been trained using the Back Propagation algorithm. The weights were updated after each pattern presentation and the learning rate and momentum were 0.2 and 0.1 respectively. The recognition rate reached 94% at 11 features with a single hidden layer containing 15 nodes.

5 CONCLUSIONS

This work presented an implementation for Arabic sign language recognition; PCNN has been used to extract static posture features and MNN as a classifier. MNN showed superiority over traditional MLP in both training time and accuracy. The system achieved more than 94% recognition accuracy on a dataset of 100 Arabic signs. Also, this work emphasizes on graph matching as a technique for recollecting the static postures to form dynamic gestures. This idea can be extended to find a gesture meaning among sequence of gestures in real time. The used techniques can be applied to other sign languages but the scientific understanding for the sign language itself is a must.

REFERENCES

- [1] J. F. Allen, B. W. Miller, E. K. Ringger, and T. Sikorski. "A robust system for natural spoken dialogue". *In Proceedings of the 34th Annual Meeting of the ACL*, pp. 62–70, 1996.
- [2] A. Samir, M. Abull-ela, Tolba, M.F., "Neutralizing lighting non-homogeneity and background size in PCNN image signature for Arabic Sign Language recognition" *Neural Computing and Applications journal*, Vol. 22, No. 1 pp. 1-7, 2012.
- [3] N. Tanibata, N. Shimada and Y. Shirai, "Extraction of Hand Features for Recognition of Sign Language Words", *Proceedings of the 15th International Conference on Vision Interface*, pp.120-128, Calgary, Canada, 2001.
- [4] F-S. Chen, C-M.Fu and C-L. Huang, "Hand gesture recognition using a real-time tracking method and hidden Markov models", *Image and Vision Computing journal*, Vol. 21, No.2, pp. 745–758, 2003
- [5] J. Zieren and K-F.Kraiss, "Non-Intrusive Sign Language Recognition For Human- Computer Interaction", *in 9th IFAC/IFIP/IFORS/IEA Symposium Analysis, Design, and Evaluation of Human- Machine Systems, Atlanta, GA*, pp. 221-228, 2004.
- [6] J. Zieren and K-F.Kraiss, "Robust Person- Independent Visual Sign Language Recognition", *Proceedings of Pattern Recognition and Image Analysis, Second Iberian Conference*, Estoril, Portugal, Vol.2, 2005.
- [7] Official Microsoft Xbox website, introduction of Kinect, <http://www.xbox.com/en-US/kinect>. "Last visit 12/7/2014"
- [8] "Integrating Speech and Hearing Challenge Individuals", YouTube channel of Dr. Natheer Khasawneh, <http://www.youtube.com/user/knatheer#p/a/u/1/vVL398dUU5Q>. "Last visit 8/7/2014"
- [9] I. K. Park, J. H. Kim, and K. S. Hong. "An implementation of an FPGA-based embedded gesture recognizer using a data glove", *Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication (ICUIMC'08)*, 2008.
- [10] Y. Shi, and T. Tsui. "An FPGA based smart camera for gesture recognition in HCI applications", *in Computer Vision-ACCV 2007*, 2007, pp. 718-727.
- [11] NVIDIA CUDATM, NVIDIA CUDA C Programming Guide Nvidia Corp. Version 4.0, May 6, 2014.
- [12] NVIDIA CUDATM, CUDA API Reference Manual, Version 4.0, February 2014.
- [13] M. Al-Rousan, O. Aljarrah and M. Hussain, "Automatic Recognition of Arabic Sign Language Finger Spelling", *International Journal of Computers and Their Applications*, Issue 1076-5204, Vol. 8, No.2. 2001, pp. 80-88.
- [14] O. Jarrah and A. Halawani, "Recognition of gestures in Arabic Sign Language using neuro-fuzzy systems", *in Artificial Intelligence*, pp. 117-138, 2001.
- [15] M.F Tolba, M. S. Abdellwahab, M. Abulle-Ela and Ahmed Samir, "Image Signature Improving by PCNN for Arabic Sign Language Recognition". *Canadian Journal on Artificial Intelligence, Machine Learning & Pattern Recognition* Vol. 1 No. 1, March 2010.
- [16] Yan Ming Zhao et al., "The Parallel Design and Implementation of the PCNN Algorithm Based on the Visual Perception Information", *Applied Mechanics and Materials*, 291-294, 2936, 2013.
- [17] J.W. McClurkin, J. A. Zarbock and L. M. Optican, "Temporal Codes for Colors, Patterns and Memories", *Cerebral Cortex*, Vol. 10, A. Peters and K. S. Rockland [eds], Plenum Press, NY, pp. 443, 1994.
- [18] H.J. Ranganath, G. Kuntimad, and J. L. Johnson, "Pulse Coupled Neural Networks for Image Processing", *PCNN International Workshop*, Huntsville, Alabama, April 5, 1995.

- [19] R. Forgá: "Feature Generation Improving by Optimized PCNN", *Applied Machine Intelligence and Informatics. SAMI 2008. 6th International Symposium on Volume*, Page(s):203 - 207, Issue, 21-22, Jan. 2008, 2008.
- [20] Y. Q. Chen, T. Yin and H. A. Babri, "A stochastic back propagation algorithm for training neural networks," *in proc. International Conference on Information, Communications and Signal Processing*, pp. 703-707, 9-12 September, 1997, Singapore.
- [21] S. C. Ng, S.H. Leung and A. Luk, "Fast convergent generalized back propagation algorithm with constant learning rate", *Neural Processing Letters*, vol. 9, pp. 13-23, 1999.
- [22] H. Qiu, E. R. Hancock: Graph partition for matching. In Proc. 4th Int. Workshop on Graph-Based Representations in Pattern Recognition, Springer LNCS2726, 178-189, 2003
- [23] M. Delalandre, E. Trupin, J. M. Ogier: Local Structural Analysis: A Primer. In Proc. Workshop on Graphics Recognition, Springer LNCS3088, 220-231, 2004.
- [24] N. Shobha Rani, «Optical Character Recognition in Application With Graph Theory And Matrix Processing», IJCSRT, 2013.

BIOGRAPHY



Prof. Dr. Mohamed Fahmy Tolba is a Professor of Scientific Computing, FCSIS (1996-Present). Dr. Tolba has more than 150 publications in the fields of AI, Image Processing, Pattern Recognition, OCR, Scientific Computing, Simulation and Modeling. Also Dr. Tolba has supervised more than 50 M.Sc. and 25 Ph.D. degrees in Ain Shams University and other Egyptian Universities.



Prof. Dr. Magdy Aboul-Ela is a professor of computer science at Sadat Academy for Management Sciences from March 2007 to present. Aboul-Ela has applied research experience in Artificial Intelligence field concerned with knowledge based Systems, Natural Language Processing, Neural Networks, and Multi-Agents Systems.



Dr. Ahmed Samir is a Lecturer at the Faculty of Computer and Information Science, Ain Shams University, Cairo, Egypt. His research interests: Image Processing and, pattern recognition and AI. He worked in Arabic Sign Language Recognition field from 2004 till now.

نموذج مقترح للتعرف علي لغة الإشارة العربية باستخدام الشبكات العصبية

¹أحمد سمير - قسم الحسابات العلمية - كلية الحاسبات و المعلومات - جامعة عين شمس
²مجدي أبو العلا - قسم علوم الحاسب - أكاديمية السادات
³محمد فهمي طلبة - قسم الحسابات العلمية - كلية الحاسبات و المعلومات - جامعة عين شمس

¹ahmed.new80@hotmail.com

²maboulela@gmail.com

³fahmytolba@gmail.com

خلاصة:

يعتبر التعرف علي لغة الإشارة من أكثر المجالات تحديا لتعامل الحاسب مع الانسان. و بالرغم من العوائق التي تمنع انتشار مترجمات لغة الإشارة فان حاجة المجتمع في تنامي مستمر. و بجانب مشاكل لغات الإشارة في دول العالم فان اللغة العربية تتمتع بمشاكل تخصها. يهدف هذا البحث لاقتراح و تنفيذ نظام سريع الرد لترجمة لغة الاشارات العربية. و قد تمت تجربة النموذج علي مئة اشارة حركية و قد تم التعرف بنجاح علي 94% منهم مع ضمان امكانية اضافة حركات. ان الجانب التقني من هذا البحث يوظف الشبكات العصبية كوسيلة للتمييز بين الاشارات المختلفة.