Journal of Theoretical and Applied Information Technology
15th September 2018. Vol.96. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

RECOGNITION OF CUNEIFORM SYMBOLS USING NEURAL NETWORK

[1]NADA A. RASHEED, [2]WESSAM LAHMOD NADOS
University of Babylon, Hillah, 51001, Iraq
E-mail: [1]nadaar@siswa.ukm.edu.my , [2]wessamlahmod@gmail.com

ABSTRACT

Recognition is regarded as a basic human attribute. In pattern recognition, there are many practical applications such as: handwriting, fingerprints, face recognition and many others. The purpose of the current work is to propose a method is able to recognize cuneiform symbols through the use of neural network. In order to achieve this objective, the image is loaded into the memory and the preprocessing can be done with introduce some modifications. Then, the attributes are extracted by finding the frequency of the on-pixels in the (50× 50) image matrix on the horizontal lines, vertical lines, the upper triangle of the main diagonal, and the lower triangle of the main diagonal. Finally, Backpropagation Algorithm was used in recognition with (20) hidden nodes and at the learning rate (0.05), within a short training time.

Keywords: *Cuneiform, Symbols, Numerals, Recognition, Backpropagation*

1. INTRODUCTION

Sumerian civilization is the earliest known civilization in the historic region of southern Mesopotamia; its location approximately identical with the modern Iraq from north of Baghdad to the Arabian Gulf as shown in Figure 1. The total area of Sumer was about 10,000 square miles [1] -[3].

Figure: Represented the Sumer location.

The Sumerian writing system [4] is one of the oldest systems of writing in the world [4][5]. In the 4th millennium BCE, the system of Sumerian writing was a Logographic [3], that are symbols or pictures represent a complete word [2]. First developed by the ancient Sumerians of Mesopotamia around 3200 BCE, it was used more than 3 millennia continuously [6]. Then, it is developed to become a cuneiform inscription on the tablets [7], which is considered one of the earliest forms of written language [2], so the tablets were used for the documentation of imported and exported goods during trade and laws [2]. As well as, record laws and important issues about agriculture, economy, politics, and so forth [8]. The clay tablets could be rectangular, circular, and cylindrical slabs. Sumerians molded the tablets manually and using a wedge-shaped stylus inscribed on the wet tablets. They kept these clay tablets in private places inside the palaces until they were dried either by the sun or by fire [9], as shown in Figure 2.

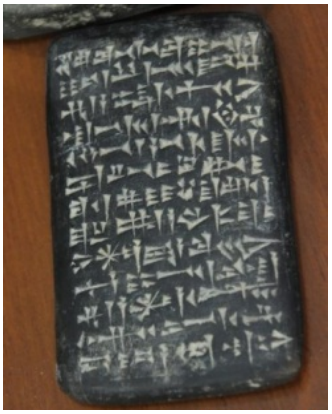


Figure 2: Represented the ancient cuneiform tablets.

Therefore, we should use computer technologies in order to translate and understand the contents of cuneiform tablets that preserved tens of thousands in the museum archives of all over the world and recognize the cuneiform

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

symbols [9][10]. Therefore, the motives for this work are listed as follows:

- Replacing manual recognition of cuneiform symbols through proposing methods that depend on the horizontal and vertical pixels, the upper and lower triangle of the main diagonal pixels.
- The reduction of human resources working to translate the cuneiform text.

Thus, the main objective of this work is to design a powerful model to recognize the symbols through the use of the neural network. To highlight the technique used in this study and the key features in order to achieve accurate matching between unknown cuneiform symbol and the cuneiform features to the database we adopted on pixel lines. That's why we used Backpropagation neural network in order to classify anonymous cuneiform symbol.

The current paper presents the methodology and experimental setup in section 2. Result and discussion are presented in section 3. Conclusions are given in section 4.

2. RELATED WORKS

The aim of this section is to review the notable studies in the body of related literature, in order to enable the researcher to develop and propose new algorithms to achieve better results than previous algorithms. Unfortunately, we have noticed that translated texts constitute only about 10% Sumerian data written [14]. Additionally, still some difficulties with Cuneiform lexicon texts to be fixed [13], [15] and few researchers have taken care of them, such as [16] implemented an algorithm to recognize the cuneiform text which depended on wavelet algorithm. Their proposed method depends on a limited number of cuneiform symbols. Which consists of two algorithms calculate the image fingerprints depending on the intensity curve of an image and search of images. They applied the method on 500 handwritten cuneiform symbols and achieved upper of 90%.

Another approach [17] had to be implemented based on 3D scanner, which includes capturing the character volume below the surface of the table and intersection with the volume of multiple spheres.

In [15], an algorithm was proposed for vertical and horizontal projections, a center of gravity, with connected component as a feature, either process of classification was carried out in two phases: the first phase, was applied K-means algorithm, and then classified the symbol within the same group by using multilayer neural networks.

The author of [18], used Online and Offline to obtain the symbol, and extract the information from of the cuneiform tablets using symbol structural vector (SSV) that achieve accuracy 97% to classify the Sumerian symbol. The authors [10] proposed a method depends on intensity profile curves to identify the cuneiform sign images.

Therefore, this work used various features through a new method to find the solution for recognition the Cuneiform Symbols.

3. METHODOLOGY & EXPERIMENTAL SETUP

In this section, we present details of our proposed method, which is designed cuneiform symbol recognition using Neural Networks. The sequence of the stages of the proposed method includes:

- ✓ Loading image.
- ✓ Preprocessing.
- ✓ Extracting features.
- ✓ Recognition using the Backpropagation algorithm.

Moreover, in this paper it uses the MATLAB application for preparing and implementing the programs to be applied. Furthermore, the algorithm was applied by using the cuneiform numerals.

2.1 Loading image

This procedure is responsible for loading .jpg format image files into memory. The image size used in this work is (100 × 100) pixels. As shown in Figure 3.

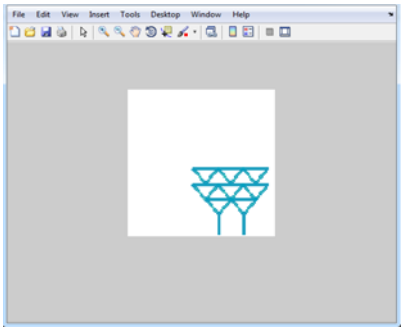


Figure 3: Image loading to the memory.

2.2 Preprocessing

The cuneiform symbol image requires some manipulation before applying recognition technique. In some cases the images have noise,

Journal of Theoretical and Applied Information Technology
15th September 2018. Vol.96. No 17

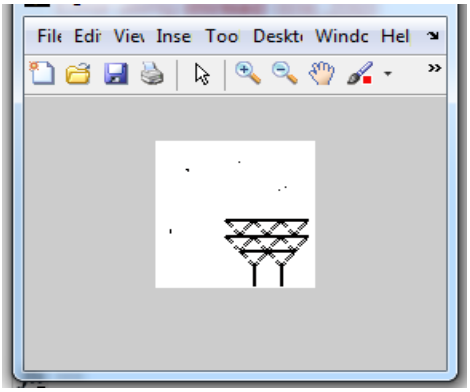
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

different colors or reflections appear small when compared with the background image. Therefore, a series of operations should be done to improve image quality [19]-[20].

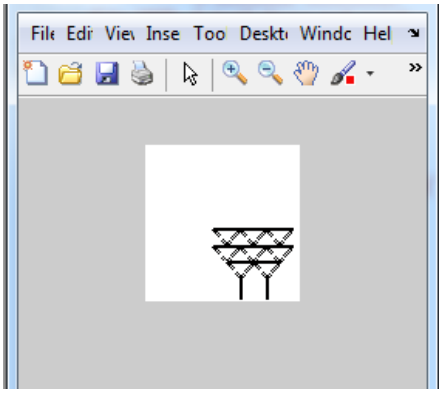
This process contributes to prepare of the image with improving the robustness in terms of features to be extracted.

2.2.1 Noise Removal

In this work is designed an algorithm to remove unwanted objects. This is done through the use of window to test the noisy objects in the image, which are within the window borders. After testing the existence of these objects and checking whether they are isolated from the cuneiform symbol, they are removed, as in the Figure 4.



(a)



(b)

Figure 4: (a) Before Noise Removal, (b) After Noise Removal.

2.2.2 Image Centralization

This procedure is important for the next step, namely image trimming, which is necessary to drag the cuneiform symbol to the center and remove the area around the center of the image. Centralization is done according to X-axis and Y-axis, see Figure 5.

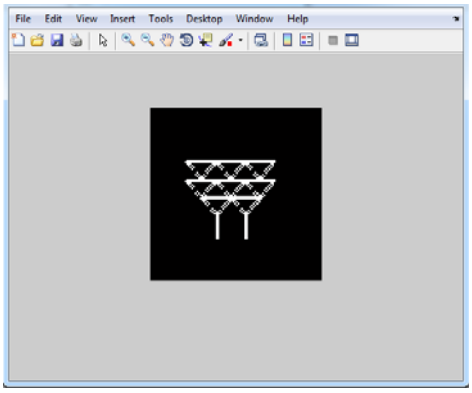


Figure 5: Image Centralization.

2.2.3 Image Trimming

Usually the image may consist the area of the cuneiform symbol and its surroundings, thus the image may include additional empty lines and columns that have no data (space lines). These empty lines should be eliminated by tracing from outside margins towards inside and stopped at the first occurrence of on-pixel on each side of the four edges, see Figure 6.

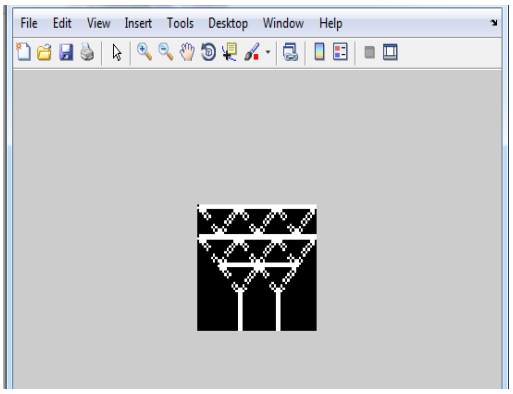


Figure 6: Image Trimming.

2.2.4 Image Scaling

In order to minimize the variation in the final results, all cuneiform symbols are scaled in fixed sizes (50×50). See Figure 7.

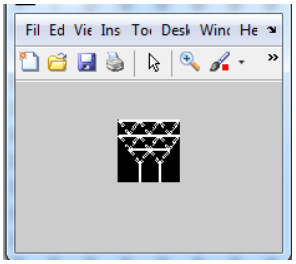


Figure 7: Image Scaling.

Journal of Theoretical and Applied Information Technology
15th September 2018. Vol.96. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2.3 Feature Extraction

Consider feature extraction is an important phase in object detection [21]. At this point, the 2D image will be converted to vector features to be input to the next step, which is the recognition symbol [16]. Locating the number feature needed for obtaining reliable cuneiform symbols is a very difficult task. It is believed that using so many features is unlikely to lead to high performance may lead to some difficulties, especially when a test symbol is compared with the reference symbols. Therefore, it must reduce the input data by extracting the characteristics that represent the most relevant of object [22]. The feature extraction stage is started by finding the histogram frequency of the on-pixels in the (50×50) image matrix, which include four histograms computed in different directions. Each histogram has (50) values and are explained as follows: Horizontal vectors computed the pixels via frequency on the horizontal lines of the image matrix. Thus, it produces a vector of (50) values. See Figure 8.

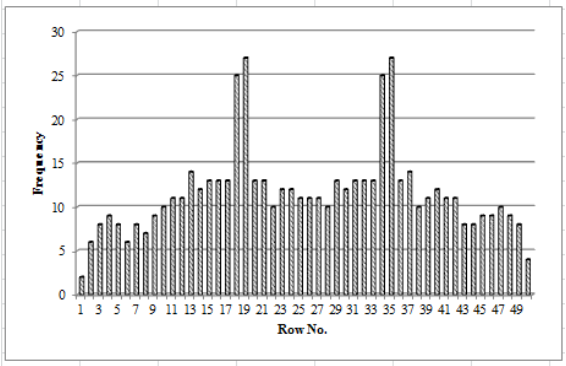


Figure 8: Feature Extraction on Horizontal Histogram.

Vertical vectors computed the pixels via frequency on the vertical lines of the image matrix. Thus, it produces a vector of (50) values. See Figure 9.

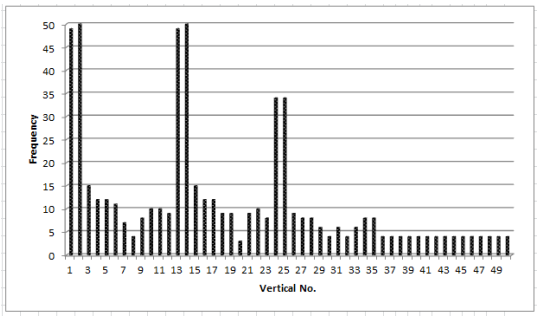


Figure 9: Feature Extraction on Vertical Histogram.

The lower triangle of the main diagonal vectors computed the pixels via frequency in the lower triangle of the main diagonal of the matrix image. Thus, it produces a vector of (50) values. See Figure 10.

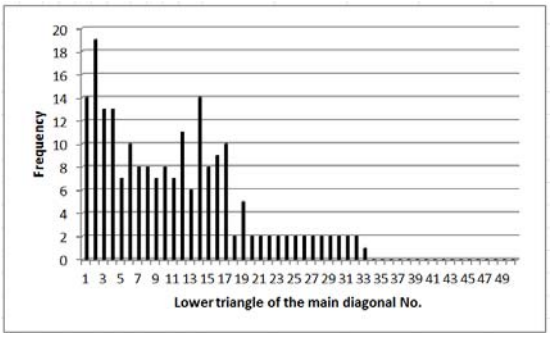


Figure 10: Feature Extraction on the lower triangle of the main diagonal.

The upper triangle of the main diagonal vectors computed the pixels via frequency in the upper triangle of the main diagonal of the matrix image. Thus, it produces a vector of (50) values. See Figure 11.

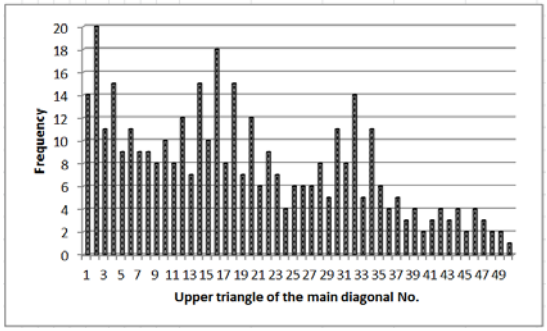


Figure 11: Feature Extraction on the upper triangle of the main diagonal.

Consequently, the result of this stage is a vector of (200) values for each symbol. Thus, the main objective of this procedure has been achieved by reducing the dimensions of the pixels of the original image, which is important to facilitate subsequent classification processes.

2.4 References Database

Most methods, but not all, during the first three phases, data acquisition, preprocessing and feature extraction, generate a reference for the cuneiform symbols.

As mentioned previously, the cuneiform symbol database preparation was used samples, which represent the cuneiform numbers (1, 2, 3, 4, 5, 6, 7, 8, 9, and 10). See Figure 12.

Journal of Theoretical and Applied Information Technology
15th September 2018. Vol.96. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

1	2	3	4	5
				
6	7	8	9	10
				

Figure 12: Represent the cuneiform numerals.

2.5 Recognition Using Backpropagation Algorithms

Classification technique is not an easy task, and several methods are used to achieve the desired results [23]. The Neural Network was used in our work, namely Backpropagation Neural Network (BNN) algorithm, which is a powerful mapping network [24] that has been applied successfully to a wide variety of problems [25]. In order to create an effective neural network, two fundamental procedures must occur, that is, feed-forward propagation and feed-backward propagation. In feed-forward propagation, one sends an input signal through the network and receives an output. Feed-backward propagation allows for the network to (learn) from its mistakes.

The first phase of implementing the feed-forward propagation sequence was intended to find an effective way to input the pattern into the network, after extracting feature vectors having (200) elements for each symbol available in the database. These vectors provide the input to the cuneiform symbol pattern recognizer, which learns the relationship among the feature vectors using the 3-layered artificial network by the Backpropagation algorithm. This model consists of (200) units in the input layer with the addition of one more unit representing the bias. The number of nodes in the second hidden layer is chosen to be (20) nodes, which were found to be more suitable for the purpose of experimentation in this problem, in addition to the bias unit as in the input layer.

Finally, the third layer is the desired output layer that is assigned one node for each side of the original input pattern, so outputs the appropriate ten-bit code arbitrarily assigned to each input pattern. Therefore, the test data will be in the range of (0, 1). Figure 13 shows the network structure:

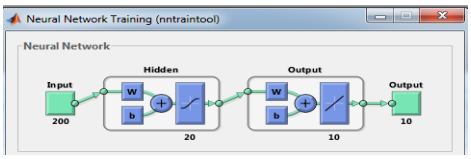


Figure 13: Diagram of a Neural Network.

Before training a neural network, it must normalize the input and network. In order to increase the network effectiveness and make it more suitable, several experiments were carried out to select the best value of the learning rate that is selected within scope of (0-1).

Therefore, this research is adopted on learning rate (0.05) that was chosen experimentally. Also, the momentum term (0.9) is used to give the best recognition results. The training continues so up the total of squared error for (10) numerals to a minimum.

4. RESULT AND DISCUSSION

Before applying the proposed algorithm for classifying cuneiform symbols, should be captured at high resolution images for each table. Additionally, should be taken care for all issues that including lighting, reflection and shadow [26] [27]. Then it must segmented each cuneiform symbol in the image, and test it separately as shown in Figure 14. After loading an image into memory, it is converted to binary image and applied the algorithms for both of noise removal and image centralization. See Figure 15.

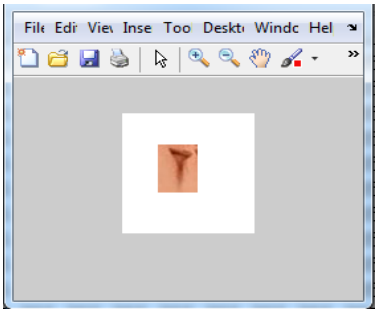
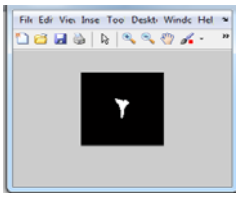


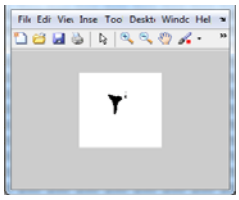
Figure 14: An image of a cuneiform symbol.



(a)

Journal of Theoretical and Applied Information Technology
15th September 2018. Vol.96. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



(b)

Figure 15: Illustrate (a) the binary image (b) noise removal and centralization of the image

Next step, implement the algorithms for trimming the empty lines and image scaling into (50× 50). See Figure 16.

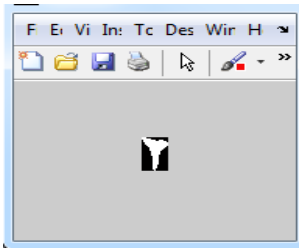


Figure 16: Trimming and Scaling Image.

Thus, calculate the four features, which include [horizontal, vertical, (lower - upper) triangle of the main diagonal] vectors. See Figures 17-20.

After creating the input matrix and target matrix in the workspace of MATLAB application, it will be developing a fitness network with (20) neurons in the hidden layer. The data for training will be 70%, and for both validation and testing will be 15%. The next step is training network and storing the desired training parameters. Actually an epoch consists of one cycle through the entire set of training vectors. Typically, number of epochs are required for training a Backpropagation neural network. The foregoing algorithm updates the weights after each training symbol is presented. A common variation is batch updating, in which weight updates are accumulated over an entire epoch (or some other number of presentations of symbols) before being applied. Thus, the plots for training will be available in Figure 21; while the training network; the update appears between the known datasets and desired outputs continuously in the window. This window is composed of the performance, the magnitude of the gradient of performance, and the amount of the validation checks. It would like to point out the training of the network that stops after the Mean Square Error (MSE) of validation data nearing the minimum, which is considered as a better performance.

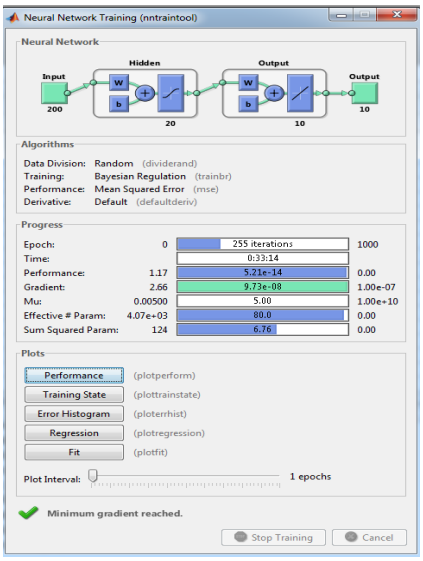


Figure 21: Neural Network Training.

The window of neural network training reflects the maximum number of training epochs, which represents the iterations to complete training. In this experiment, although assuming (1000) epochs, network training has been completed only about (255) of iterations in that an epoch is one cycle through the entire set of training vectors, the weights are updated during each epoch until the error between the output and target is to the minimum. From the train window, it can access the regression plot that demonstrates the relationship between the outputs of the network and the targets in four plots (training, validation, test, and all performances).

Here the training R is (0.99), and for all, it is (0.90). Therefore, this indicates that there is an exact linear relationship between outputs and targets. After training the Backpropagation neural network, it should recognize the unknown symbol by applying the feed-forward phase algorithm using the data of the test symbol. Therefore, the results illustrate in the Table 1.

Table 1: The results of the test symbol

NO. of Nod	Results of The Test Nodes
1	0.7
2	0.2
3	-0.1
4	-0.2
5	0.2
6	0.5
7	0.4
8	0.2
9	0.6
10	0.4

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

In the table, we observe that unknown symbol achieved highest value (0.7), which is represented in the first node. That means the unknown symbol indicates to the number one, and this indicates to the success of the proposed method, and this fact leads us to the conclusion that cuneiform texts can be translated by adding more symbols to the database. Then the data of an unknown symbol are entered to distinguish it through assisting the computer.

The objective to learn the classification from sample data is to classify new data set to predict successful study. However, in most practical pattern recognition formulation, a classification rule does not lead to perfection. There always a probability of classification error, that is, with type I (FRR: False Rejection Rate) and type II (FAR: False Acceptance Rate) error rates. Through applied experiments were performed on (10) test symbols that shown in Figure 22. The proposed method accepts (8) symbols and refuses (2) so type I error is (2%). The results illustrate in the Tables II-XI.

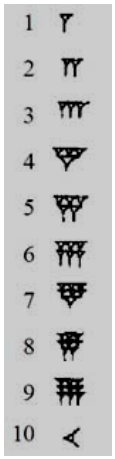


Fig. 22. Illustrate the Sumerian numbers that used in the test.

Table 2: The results of the test number one

NO. of Nod	Results of The Test Nodes
1	0.99411
2	0.0024031
3	0.65555
4	0.0098623
5	0.092466
6	0.00014946
7	0.0023841
8	0.097534
9	0.076306
10	0.11873

Table 3: The results of the test number two

NO. of Nod	Results of The Test Nodes
1	0.064882
2	0.97928
3	0.59873
4	0.00084501
5	0.8905
6	0.67389
7	0.020488
8	0.03226
9	0.00038088
10	0.75446

Table 4: The results of the test number three

NO. of Nod	Results of The Test Nodes
1	0.099672
2	0.0049905
3	0.88468
4	0.013899
5	0.199257
6	0.85496
7	0.16633
8	0.079951
9	0.9991
10	0.29971

Table 5: The results of the test number four

NO. of Nod	Results of The Test Nodes
1	0.0043823
2	0.00051607
3	0.0035829
4	0.90944
5	0.79643
6	0.070317
7	0.011637
8	0.39486
9	2.4825e-05
10	0.0069697

Table 6: The results of the test number five

NO. of Nod	Results of The Test Nodes
1	0.10571
2	0.00011457
3	0.037363
4	0.0003581
5	0.94139
6	0.48367
7	0.0092192
8	0.15305
9	0.75555
10	0.49083

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 7: The results of the test number six

NO. of Nod	Results of The Test Nodes
1	0.72701
2	4.894e-05
3	0.057989
4	0.0022022
5	0.1686
6	0.98972
7	0.053066
8	0.10108
9	0.51581
10	0.65159

Table 8: The results of the test number seven

NO. of Nod	Results of The Test Nodes
1	0.9353
2	0.0063757
3	0.21453
4	0.013176
5	0.90404
6	0.89154
7	0.98476
8	0.84655
9	0.99993
10	0.049422

Table 9: The results of the test number eight

NO. of Nod	Results of The Test Nodes
1	0.048226
2	0.0001877
3	0.24007
4	0.010149
5	0.63757
6	0.75481
7	0.032328
8	0.99533
9	0.16569
10	0.3565

Table 10: The results of the test number nine

NO. of Nod	Results of The Test Nodes
1	0.34019
2	0.00033456
3	0.61501
4	0.033235
5	0.70539
6	0.31571
7	0.081485
8	0.30422
9	0.9996
10	0.29426

Table 11: The results of the test number ten

NO. of Nod	Results of The Test Nodes
1	0.70015
2	0.0046616
3	0.59845
4	0.1184
5	0.88182
6	0.38751
7	0.066907
8	0.65202
9	0.49999
10	0.99954

5. CONCLUSIONS

In this work, we extracted features that included four histograms (Horizontal lines, Vertical lines, Upper triangle of the main diagonal, and the lower triangle of the main diagonal), which gave a good result. Where the algorithm has been applied to a standard Sumerian symbol database, and the results achieved 80% that is demonstrated promising result. It can be concluded that this methodology can used for recognizing other objects, because it is effective and achieves promising results. That means maybe we can use and apply the same idea of the other patterns for recognition. The proposed methodology has proven its effectiveness to classify by applying Nural Network, especially the Backpropagation network, which proved the effectiveness in pattern recognition by comparing our result of different NN architectures, and it was found that the network with (20) hidden nodes produce good results, which means there is need to vary the number of the hidden nodes to determine the number of hidden neurons for the network in order to perform its best work.

REFRENCES:

[1] S. N. Kramer, *The Sumerians, Their History, Culture, and Character*, The University of Chicago, printed in the United States of America, 1963, pp. 1-372.

[2] K. Kuiper, Mesopotamia the World's Earliest Civilization, Britannica Educational Publishing, New York, 2011, p.40.

[3] G. Zólyomi, An Introduction to The Grammar of Sumerian, Executive publisher: the Dean of the Faculty of Humanities of Eötvös Loránd University, Editorial manager: Ádám Gaborják, Budapest, 2017, p.15.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[4] D. A Foxvog, Introduction to Sumerian Grammar, Guerneville, California, USA, 2016, p.5.

[5] D. A. AL-Nasrawi, H. F. AL-Shahad, I. R. Shareef, "Characters Map for Cuneiform Writing System," *The Forth Scientific Conference of the College of Science*, University of Kerbala, Journal of University of Kerbala, 2016, pp. 195-204.

[6] S. D. Houston, The Shape of Script: How and Why Writing Systems Change. Santa Fe, New Mexico: School for Advanced Research Press, 2012, p.3.

[7] Y. Bloch, L. A. Peri, *The Israel Museum, Jerusalem Israel Museum Studies in Archaeology*, 8, 2017, p.4.

[8] E. Uchida and R. Watanabe,"Blackening of the Surfaces of Mesopotamian Clay Tablets Due to Manganese Precipitation," *Archaeological Discovery*, vol. 2, 2014, pp. 107-116.

[9] S. I. Woolley, N. J. Flowers, T. N. Arvanitis, A. Livingstone, T. R. Davis, J. Ellison, 3D Capture Representation and Manipulation of Cuneiform Tablets, SPIE Proceedings IST/SPIE Electronic Imaging, 4298, pp.103 – 110, 2001.

[10] R. Majeed, Z. Beiji, H. Hatem and J. Waleed, "Ancient Cuneiform Text Extraction Based on Automatic Wavelet Selection," *International Journal of Multimedia and Ubiquitous Engineering*, vol.10, no.6, 2015, pp.253-264.

[11] É. Pagé-Perron, M. Sukhareva, I. Khait and C. Chiarcos, Machine Translation and Automated Analysis of the Sumerian Language, Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature. Proceedings, Vancouver, BC, 2017, pp. 10-16.

[12] J. A. Bowen, A Preliminary Study of The Sumerian Curricular and Lamentational Texts From The Old Babylonian City of Kish, Thesis, Baltimore, Maryland, 2017, pp.1-445.

[13] H. Yousif, A. M. Rahma and H. Alani, "Cuneiform Symbols Recognition Using Intensity Curves," *The International Arab Journal of Information Technology*, vol. 3, no. 3, 2006, pp.237-241.

[14] É. Pagé-Perron, M. Sukhareva, I. Khait, C. Chiarcos, Machine Translation and Automated Analysis of the Sumerian Language, Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature. Proceedings, Vancouver, BC, pp. 10–16, 2017.

[15] N. M. Edan, "Cuneiform Symbols Recognition Based on K-Means and Neural Network,"Raf. J. of Comp. & Math's. , vol. 10, no. 1, *Fifth Scientific Conference Information Technology*, 2013, pp. 195-202.

[16] F. Yang, M. Hamit, C. B. Yan, J. Yao, A. Kutluk, X. M. Kong, and S. X. Zhang,"Feature Extraction and Classification on Esophageal X-Ray Images of Xinjiang Kazak Nationality," *Hindawi, Journal of Healthcare Engineering*, vol. 2017, 2017, pp. 1-11.

[17] H. Mara, S. Krömker, S. Jakob and B. Breuckmann, "GigaMesh and Gilgamesh – 3D Multiscale Integral Invariant Cuneiform Character Extraction," *Proceeding of the 11th International Symposium on Virtual Reality*, Archaeology and Cultural Heritage VAST, 2010, pp.131-138.

[18] K. K. Ahmed, Online Sumarians Cuneiform Detection Based on Symbol Structural Vector Algorithm, (2012) *J. of College of Education for Women*, 23 (2), pp. 545-553.

[19] P. K. Gowda, S. Chethan, J. Harsha, J. Rakesh, K. N. Tanushree, Offline Kannada Handwritten Word Recognition Using Locality Preserving, (2017), *International Journal of Innovative Research in Computer and Communication Engineering Projections (LPP)*, 5(5), pp.9955-9960.

[20] M. R. Barkul, S. S. Lokhande, A Review: Handwritten Character Recognition System, (2016), *International Journal of Advanced Research in Computer and Communication Engineering*, 5(2), pp. 262-265.

[21] P. Chouhan, M. Tiwari, Feature Extraction Techniques for Image Retrieval Using Data Mining and Image Processing Techniques, (2016), *International Journal of Advanced Research in Computer and Communication Engineering*, 5(5), pp.530-535.

[22] A. Mars, G. Antoniadis, Arabic Online Handwriting Recognition Using Neural Network, (2016), *International Journal of Artificial Intelligence and Applications (IJAIA)*, 7(5), pp.51-59.

[23] H. A. Alwzwazy, H. M. Albehadili, Y. S. Alwan, N. E. Islam, Handwritten Digit Recognition Using Convolutional Neural Networks, (2016), *International Journal of Innovative Research in Computer, and Communication Engineering*, 4(2), pp. 1101-1106.

[24] N. A. Rasheed, M. J. Nordin, A. H. Dakhee, W. L. Nados and M. K. A. Maaroof," Classification Archaeological Fragments into

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Groups," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 14, no.9, 2017, pp. 324-333.

[25] R. O. Duda, P. E. Hart and D. G. Stork, Pattern classification. Edition: 2nd, Chapter 6, Wiley-Interscience publication, 2001, pp. 10-74.

[26] N. A. Rasheed, M. J. Nordin, A Polynomial Function in the Automatic Reconstruction of Fragmented Objects, (2014), *Journal of Computer Science*, 10 (11), pp. 2339-2348.

[27] R. Shweka, Y. ChouekaLior, L. Wolf, N. Dershowitz, Automatic extraction of catalog data from digital images of historical manuscripts, (2013), *LLC Journal*, 28(2), pp. 315-330.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

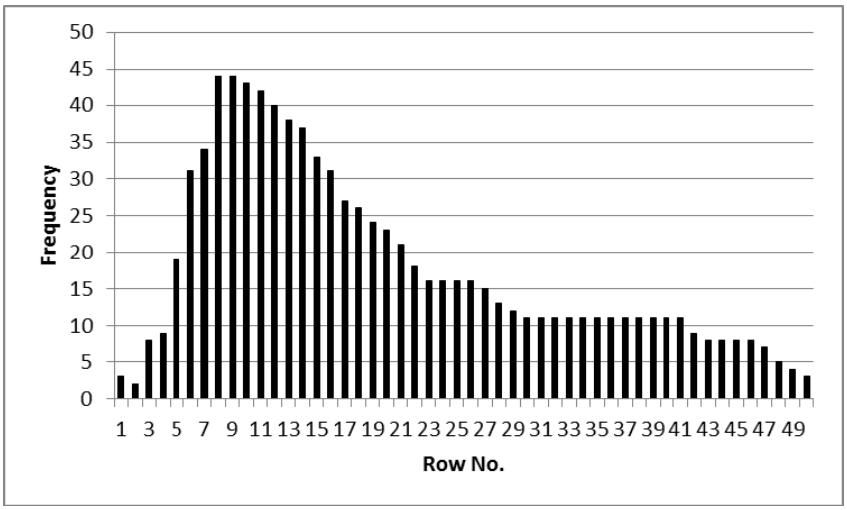


Figure 17: Feature Extraction on Horizontal Histogram for the test symbol.

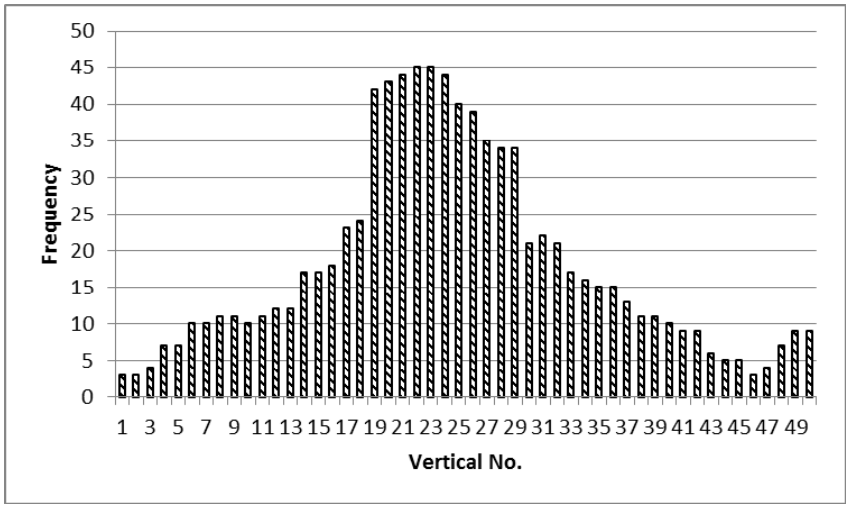


Figure 18: Feature Extraction on Vertical Histogram for the test symbol.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

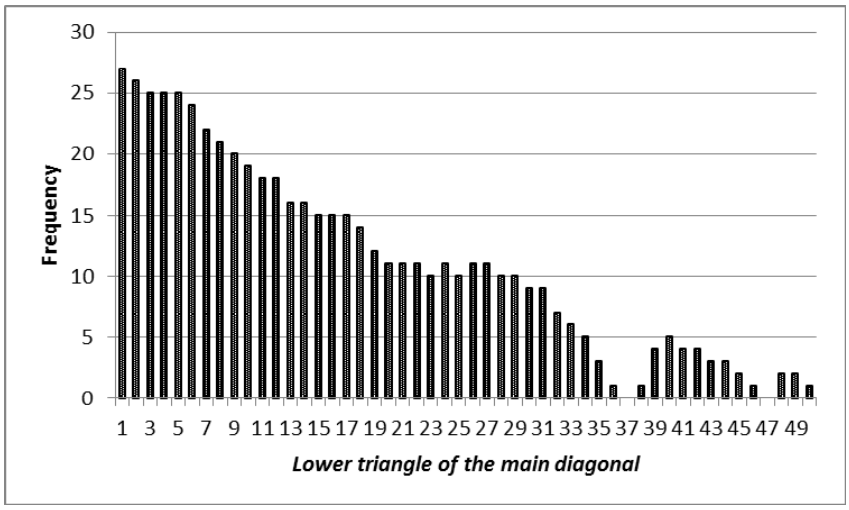


Figure 19: Feature Extraction on the lower triangle of the main diagonal for the test symbol.

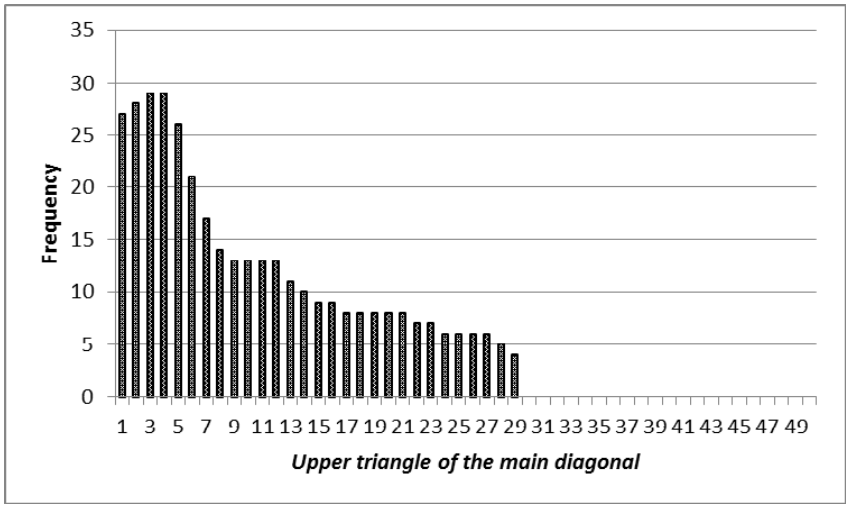


Figure 20: Feature Extraction on the upper triangle of the main diagonal for the test symbol.