# Intelligent Recognition of Ancient Persian Cuneiform Characters

Fahimeh Mostofi[1,2] and Adnan Khashman[1]
[1]Intelligent Systems Research Center, Near East University, Near East Boulevard, Lefkosa, Mersin 10, Turkey
[2]Department of Computer Engineering, Near East University, Near East Boulevard, Lefkosa, Mersin 10, Turkey

Keywords: Ancient Script, Persian Cuneiform Language, Character Recognition, Image Processing, Neural Networks.

Abstract: This paper presents an intelligent character recognition system based on utilising a back propagation neural network model. The characters in question are unique and rare to be addressed in such applications. These are the ancient Persian Cuneiform alphanumerical characters. The recognition system comprises firstly, image processing phase where clear and noisy or degraded images of the ancient script are prepared for processing by the neural model in the second phase. The importance of such application lies in its potential to make translating ancient scripts and language easier, faster and more efficient. Experimental results indicate that our proposed method can be further applied successfully to other ancient languages and may be utilised in museums and similar environments.



## 1 INTRODUCTION

Being able to read and decipher ancient scripts has always been of much interest and importance for human-being. Many valuable historical secrets were revealed by archaeologists through the time, made us aware of the culture and civilization of our predecessor.

The ancient near east was the home of early civilizations such as Mesopotamia, ancient Egypt, ancient Iran, Anatolia, Levant etc.

The Achaemenid Persian empire was the largest that the ancient world had seen, extending from Anatolia and Egypt across western Asia to northern India and Central Asia. Its formation began in 550 B.C (The Achaemenid Persian Empire, 2000).

The Old Persian language appears in royal inscriptions, written in a specially adapted version of cuneiform.

Old Persian cuneiform is a semi-alphabetic cuneiform script that was the primary script for the Old Persian language. Texts written in this cuneiform were found in Persepolis, Susa, Hamadan, Armenia, and along the Suez Canal (Kent 1950).They were mostly inscriptions from the time period of Darius the Great and his son Xerxes the Great kings of Achaemenid Empire (Kent, 1950).

Application of Artificial Neural Networks (ANN) for pattern recognition and character recognition has been more widely reported in literature recent times. This has led to high expectation of what neural networks can do for different fields, especially fields where other approaches have not been successful (Kashyap et al., 2003).

Artificial Neural network is an information-processing unit that is much inspired by the way the human brain works. Brain can do some computation (such as pattern classification and recognition) faster than conventional computers (Patra et al., 2010).

Many researchers have been working on scripts recognition for more than three decades. Nevertheless, it remains to be one of the most challenging problems in pattern recognition (Kashyap et al., 2003).

Reading cuneiform symbols is an important subject for understanding cuneiform tablets contents. Where there are a great number of cuneiform tablets representing valuable information of ancient history.

Yousif H. and et al proposed a method that used the intensity profile curves for selected pixels in images of Cuneiform text to differentiate between them (Yousif et al., 2006).

In 2013 Naktal M. Edan proposed a method for Cuneiform symbol recognition which was using a k-mean algorithm for clustering similar Cuneiform symbols and then classify symbols within the same cluster by using a multilayer neural network (Edan, 2013).

In this paper we propose a multilayer feed-

Mostofi F. and Khashman A..
Intelligent Recognition of Ancient Persian Cuneiform Characters.
DOI: 10.5220/0005035401190123
In *Proceedings of the International Conference on Neural Computation Theory and Applications* (NCTA-2014), pages 119-123
ISBN: 978-989-758-054-3

forward neural network based classification technique for recognition of Old Persian cuneiform characters.

The paper is organized as follows: Section 2 describes Old Persian Cuneiform alphabet and its characteristics. Section 3 elucidates the general architecture of old Persian Cuneiform classification system. In section 3, image preparation step has been described which explains how we supply training and testing dataset. Section 4 explains which image preprocessing methods have been used. In section 5, we describe the design of multilayer feed forward neural network for this application and finally in section 6, results of the application are discussed.

## 2 DESCRIBE OLD PERSIAN CUNEIFORM ALPHABET

Most scholars consider old Persian writing system to be an independent invention because it has no obvious connections with other writing systems at the time, such as Elamite, Akkadian, Hurrian, and Hittite cuneiforms (Windfuhr, 1970). While Old Persian's basic strokes are similar to those found in cuneiform scripts, Old Persian texts were engraved on hard materials, so the engravers had to make cuts that imitated the forms easily made on clay tablets (Kent, 1950).

The signs are composed of horizontal, vertical, and angled wedges. There are four basic components and new signs are created by adding wedges to these basic components (Windfuhr, 1970).

These four basic components are two parallel wedges without angle, three parallel wedges without angle, one wedge without angle and an angled wedge, and two angled wedges (Windfuhr, 1970).

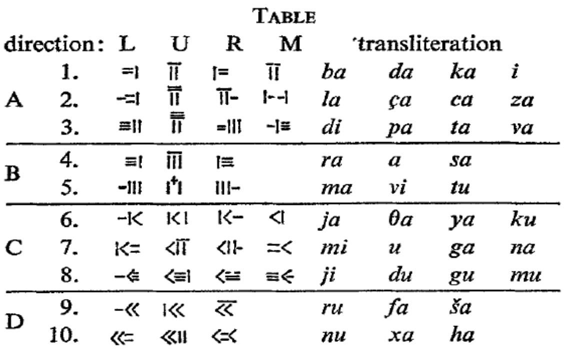


Figure 1: Old Persian Cuneiform symbols consist of four basic components (Kent, 1950).

The script is written from left to right (Daniels and Bright, 1996).

Old Persian cuneiform alphabet set splits into set of independent vowels (3 characters), constant letters (33 characters), special signs such as signs for country God and the king (8 characters), punctuation (1 character) and numbers (5 characters).In conclusion there are totally 50 separated signs for old Persian cuneiform alphabet.


Figure 2: Old Persian Cuneiform Symbols (The Unicode consortium v6.3).

## 3 THE PROPOSED SYSTEM ARCHITECTURE

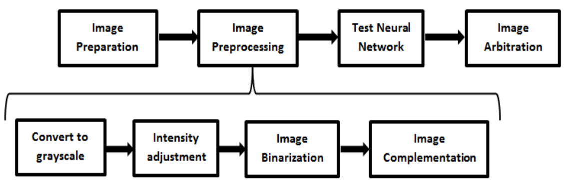


Figure 3: The proposed system architecture for Old Persian character recognition.

Old Persian character recognition system consists of five main steps: Image preparation, image pre-processing, neural network design and training, character recognition.

## 4 IMAGE PREPARATION

As we mentioned in previous section Old Persian alphabet has totally 50 signs. In this paper we have omitted special signs (such as king, country etc) from the database. Instead in addition to the original numbers which are 1, 2, 10, 20 and 100 we have added numbers 3, 4, 30 and 40 in our dataset.

Thus finally our training set consists of 3 independent vowels, 33 constants, 9 numbers (1,2,3,4,10,20,30,40,100) and 1punctuation character which are totally 46 characters. Therefore the training set in this paper consists of 46 jpg images by size 64*64 pixels each is showing an old Persian cuneiform alphanumeric character.

| | Old Persian | Latin | New Persian | Unicode | | Old Persian | Latin | New Persian | Unicode |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 𐎠 | A | آ | 103A0 | 24 | 𐎷 | MI | می | 103B7 |
| 2 | 𐎡 | I | ای | 103A1 | 25 | 𐎸 | MU | مو | 103B8 |
| 3 | 𐎢 | U | او | 103A2 | 26 | 𐎹 | YA | یَ | 103B9 |
| 4 | 𐎣 | KA | کَ | 103A3 | 27 | 𐎺 | VA | وَ | 103BA |
| 5 | 𐎤 | KU | کو | 103A4 | 28 | 𐎻 | VI | وی | 103BB |
| 6 | 𐎥 | GA | گَ | 103A5 | 29 | 𐎼 | RA | رَ | 103BC |
| 7 | 𐎦 | GU | گو | 103A6 | 30 | 𐎽 | RU | رو | 103BD |
| 8 | 𐎧 | XA | خَ | 103A7 | 31 | 𐎾 | LA | لَ | 103BE |
| 9 | 𐎨 | CA | چَ | 103A8 | 32 | 𐎿 | SA | سَ | 103BF |
| 10 | 𐎩 | JA | جَ | 103A9 | 33 | 𐏀 | ZA | زَ | 103C0 |
| 11 | 𐎪 | JI | جی | 103AA | 34 | 𐏁 | SHA | شَ | 103C1 |
| 12 | 𐎫 | TA | تَ | 103AB | 35 | 𐏂 | SSA | تر | 103C2 |
| 13 | 𐎬 | TU | تو | 103AC | 36 | 𐏃 | HA | هَ | 103C3 |
| 14 | 𐎭 | DA | دَ | 103AD | 37 | 𐏑 | 1 | 1 | 103D1 |
| 15 | 𐎮 | DI | دی | 103AE | 38 | 𐏒 | 2 | 2 | 103D2 |
| 16 | 𐎯 | DU | دو | 103AF | 39 | 𐏒𐏑 | 3 | 3 | - |
| 17 | 𐎰 | THA | ثَ | 103B0 | 40 | 𐏒𐏒 | 4 | 4 | - |
| 18 | 𐎱 | PA | پَ | 103B1 | 41 | 𐏓 | 10 | 10 | 103D3 |
| 19 | 𐎲 | BA | بَ | 103B2 | 42 | 𐏔 | 20 | 20 | 103D4 |
| 20 | 𐎳 | FA | فَ | 103B3 | 43 | 𐏔𐏓 | 30 | 30 | - |
| 21 | 𐎴 | NA | نَ | 103B4 | 44 | 𐏔𐏔 | 40 | 40 | - |
| 22 | 𐎵 | NU | نو | 103B5 | 45 | 𐏕 | 100 | 100 | 103D5 |
| 23 | 𐎶 | MA | مَ | 103B6 | 46 | 𐏐 | divider | واژه جدا کن | 103D0 |

Figure 4: Old Persian characters which are included in the training and testing dataset.

For creating a test dataset we have applied Gaussian Filter in original images. Gaussian filtering is used to blur images and remove noise and details.in this paper we have used Gaussian filter to remove some details from images and make them look blurred.

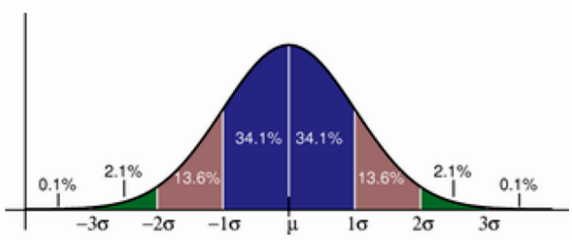


Figure 5: Gaussian Function Distribution (Gaussian Filtering 2014).

The Standard deviation (σ) of the Gaussian function plays an important role in its behaviour. The values located between +/- 3σ account for about 99% of the set. Larger values of σ produce a wider peak (greater blurring) (Gaussian Filtering 2014).

We have tested the neural network by adding different levels of noise to images. Setting σ to 3, 3.5 and 4 made the images blurred in three different levels.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{-x^2+y^2}{2\sigma^2}} \quad (1)$$

The equation (1) indicates Gaussian function where $x$ is the distance from the origin in the horizontal axis, $y$ is the distance from the origin in the vertical axis, and $\sigma$ is the standard deviation of the Gaussian distribution (Nixon, Aguado 2002).

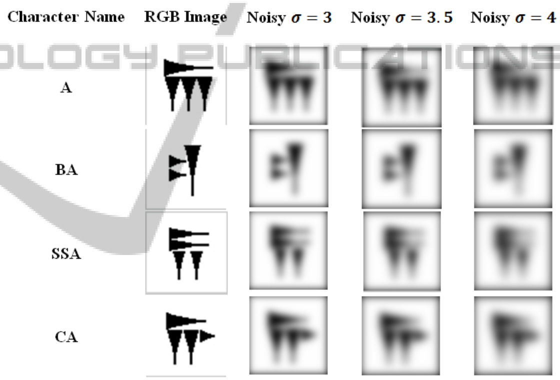


Figure 6: Demonstration of three different levels of Gaussian noise applying to four Old Persian characters.

In figure 6 images in the first row indicates firstly the original RGB image of character A and then noisy images with Gaussian filter with $\sigma = 3$ and then $\sigma = 3.5$ and finally $\sigma = 4$.increasing value of $\sigma$ produce more blurred images.the next third rows are showing the same blurring process for characters BA,SSA and CA.

## 5 IMAGE PREPROCESSING

By using image preprocessing techniques and thus reducing image data, we can increase the speed of learning by reducing computation expense.

In this section image preprocessing steps are explained.

Firstly the RGB images are converted to

grayscale, so that pixel values are in range 0 to 255 after conversion.

Secondly intensity adjustment technique has been used for increasing image contrast, which maps the image intensity values to a new range.In this paper this task was accomplished by remapping the data values to fill the entire intensity range [0, 255]. Thirdly for characters to be recognized efficiently we need a suitable binarization algorithm which can separate characters from the background accurately. The binarization arithmetic is shown in Eq.2, where $f(i,j)$ is the original character image, $f_b(i,j)$ is the binarized image and T is the threshold. Otsu's method has been used for determining T.

$$f_b(i,j) = \begin{cases} 1 & f(i,j) \geq T \\ 0 & f(i,j) < T \end{cases} \quad (2)$$

Finally In order to make the training operation faster we have complemented images.in this step the images are binary so complementation is possible by subtracting the image from 1.

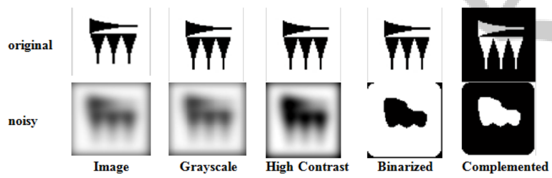


Figure 7: Four steps of image preprocessing applied on one original image in first row and the noisy image in second row.

## 6 NEURAL NETWORK DESIGN

Among many neural networks arithmetic in character recognition, back propagation (BP) arithmetic is the most robust one.

The main principles of back propagation neural networks can be concluded as following.

1) Input the character image into the BP neural networks.

2) Calculate the Error Function by comparing the recognizing result of the neural networks and the known character result so as to adjust the value of the connective parameters between layers(weights) and make the output of the network more approximate to the known result.

3) Train the networks with a series of known images with the purpose of optimizing the parameters of the networks (Yang,Yang, 2008).

In this paper old persian cuneiform characters were classified by using back propagation neural network The designed neural network consists of three layers: Input Layer, hidden layer and output layer.the number of neurons in input layer is 4096 which represents total number of pixels of a sample image.the number of neurons in output layer is 46 which refers to the total number of Old Persian character samples, and the number of neurons in hidden layer is determined by experiment which is 44 in this neural network.

For both hidden and output layer, log-sigmoid activation function has been used. As it has been shown in table below Network Training parameter epoch has been set to 5000 and training goal was 0.005.Training was stopped when number of epoch is exceed or minimum error is meet. Momentum rate was set to 0.45 while learning coefficient has been set to 0.002.

The neural network is trained using 46 Old Persian cuneiform character images. The 46 training images are noise-free and are shown in examples in Figure 3. The remaining images contain: 46 images with $\sigma = 3$ noise level, 46 images with $\sigma = 3.5$ noise level and 46 images with $\sigma = 4$ noise level .these are the testing images which are not used in training set and shall be used to test the robustness of the trained neural network in identifying the characters.

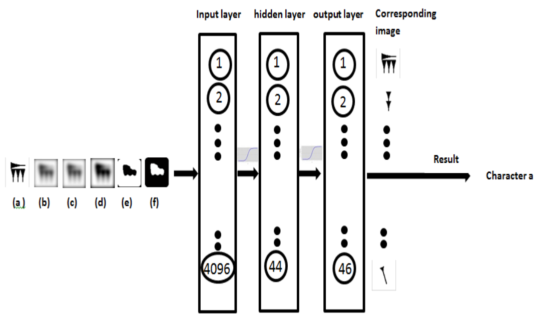


Figure 8: Old Persian cuneiform character recognition neural network topology

## 7 RESULT AND DISCUSSION

Table 1 lists the final parameters of the successfully trained neural network, which learnt after 243 iterations and within 13 seconds. The running time for the neural network after training and using one forward pass was 0.07 seconds; this running time is the time required for the trained neural network to identify one character.

The old persian cuneiform character recognition system was tested for three set of noisy images and As it is shown in Table recognition rate for 46 images of training set was 100% correct identification as it would be expected.

For the test set we applied three different levels of Guassian filter

- By applying Guassian filter with $\sigma = 3$ To 46 images 100% (46 out of 46) of Noisy images were identified correctly.
- By applying Guassian filter with $\sigma = 3.5$ To 46 images 93.4% (43 out of 46) of Noisy images were identified correctly.
- By applying Guassian filter with $\sigma = 4$ To 46 images 89.1% (41 out of 46) of Noisy images were identified correctly.

Table 1: Neural network parameters.

| Parameter | Value |
|---|---|
| Input Neurons | 4096 |
| Hidden Neurons | 44 |
| Output Neurons | 46 |
| Learning Coefficient | 0.002 |
| Momentum Rate | 0.45 |
| Minimum Error | 0.005 |
| Maximum Iterations | 5000 |
| Iterations for training | 243 |
| Training Time(seconds) | 13 |
| Run Time(seconds) | 0.07 |

Table 2: Neural network system's recognition rate.

| **Noise Level** | $\sigma = 3$ | $\sigma = 3.5$ | $\sigma = 4$ |
|---|---|---|---|
| TrainSet | 100% | 100% | 100% |
| TestSet | 100% | 93.4% | 89.1% |

# 8 CONCLUSION

This paper presented Old Persian Cuneiform character recognition system based on artificial neural network.

The training database of this system consisted of 46 noise-free images of 46 alphanumeric old Persian characters.

Applying three different levels of Guassian filter with $\sigma = 3$, $\sigma = 3.5$ and $\sigma = 4$ to the original images lead to creating a test dataset with totally 184 images.(46 noise-free images and 46 images for each particular level of Gaussian filter.)

Training the neural network uses only 46 noise-free images and testing the trained neural network was performed using four different set of images with different levels of noise.

The highest obtained rate for correct identification of testing set Old Persian character images was 100%. These images were not feed to the neural network during training.

# REFERENCES


Daniels, P., Bright, W., 1996. The World's Writing Systems, Oxford University Press.

Edan , N., 2013. Cuneiform symbols recognition based on k-means and neural network, *fifth scientific conference information technology*, December 19-20.

Gaussian filtering lecture handouts of the University of Auckland Department of Computer Science, 2014, Accessed 21 April 2014, Available from: <https://www.cs.auckland.ac.nz/courses/compsci373s1c/PatricesLectures/Image%20Filtering.pdf>

Kashyap, k., Bansilal, Koushik, P., 2003. Hybrid neural network architecture for age identification of ancient kannada scripts,*circuits and systems,2003.ISCAS '03.Proceeding of the 2003 international symposium on.(volum:5)*, 25-28 May.

Kent, R., 1950. Old Persian Grammar Texts Lexicon, American Oriental Society, New Haven.

Nixon, M., Aguado, A., 2002. Feature Extraction and Image Processing. Newnes, 1st edition.

Patra, P., Vipsita, S., Mohapatra, S., Dash, S., 2010. A novel approach for pattern recognition, *International Journal of Computer Applications (0975 – 8887) Volume 9– No.8*, November 2010.

The Achaemenid Persian Empire,2000 , Accessed 21 April 2014, available from <http://www.metmuseum.org/toah/hd/acha/hd_acha.htm >

The Unicode consortium, v6.3, Accessed 21 April 2014, Available from <http://www.unicode.org/charts/PDF/U103A0.pdf>

Windfuhr, G., 1970. Notes on the old Persian signs, Indo-Iranian Journal XII/2, pp. 121-125.

Yang, F., Yang, F., 2008. Character recognition using parallel BP network, *Audio language and image processing 2008.ICALIP.*, shanghai, July 7-9 .

Yousif, H., Rahma, A., Alani, H., 2006. Cuneifrom symbols recognition using intensity curves, *The international Arab journal of information technology vol.3,No.3* ,July 2006.