A CASE STUDY IN COMPUTER UNDERSTANDING OF PRINTED-FORMS

Davood Falahati [1], Hojat Cheraghi [2] and Kazem Ghalamchi[3]

[1]Department of Electrical Engineering,
Isfahan University of Technology, Isfahan-Iran
d.falahati.1987@ieee.org
[2]Tehran Science and Research University, Tehran-Iran
hojat.ch@gmail.com
[3]Ghalamchi Foundation,Tehran, Iran.
kazemglmchi@yahoo.com

ABSTRACT

Data entry is a time consuming and erroneous procedure in its nature. In addition, validity check of submitted information is not easier than retyping it. In a mega-corporation like Kanoon Farhangi Amoozesh, there are almost no way to control the authenticity of students' educational background. By the virtue of fast computer architectures, optical character recognition, a.k.a. OCR, systems have become viable. Unfortunately, general-purpose OCR systems like Google's Tesseract are not handful because they don't have any a-priori information about what they are reading. In this paper the authors have taken a in-depth look on what has done in the field of OCR in the last 60 years. Then, a custom-made system adapted to the problem is presented which is way more accurate than general purpose OCRs. The developed system reads more than 60 digits per second. As shown in the Results section, the accuracy of the devised method is reasonable enough to be exposed in public use.

KEYWORDS

Optical character recognition, tesseract, neural networks, row finding, segmentation.

1. INTRODUCTION

Data-entry phase, is by far the most time-consuming and the deadliest line of work in data acquisition process. A remarkable portion of a company's human resource should be devoted to collect printed information in the forms. Computer-assisted data-entry process has been a human ancient dream. In the latest sixty years, there have been exerted massive efforts to implement an automatic character recognition system [1]. Template-Matching was one of the earliest methods for character recognition in which an unknown character should be compared to all of the possible candidates. Optical imaging techniques were the backbone of identification systems before 70s. Hongo and Nitta devised an optical system that processed a video signal using template matching [2].

David C. Wyld et al. (Eds) : CCSIT, SIPP, AISC, NLP - 2015
pp. 53–62, 2015. © CS & IT-CSCP 2015 DOI : 10.5121/csit.2015.50405

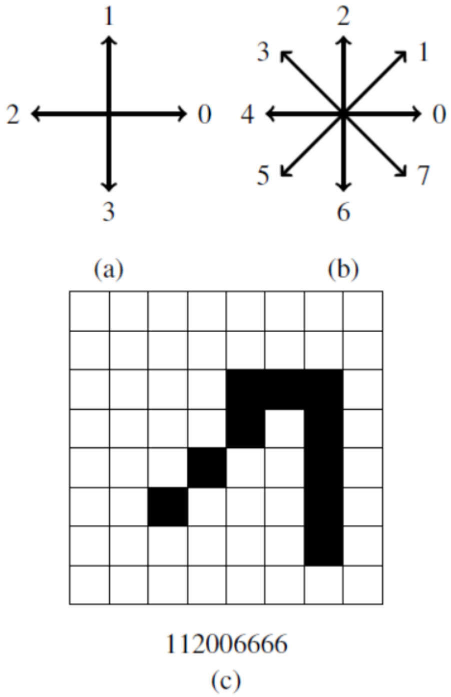

Fig. 1. Chain codes grid. (a) 4-connected gird (b) 8-connected grid. (c) a sample of coded sequence using 8-connected grid.

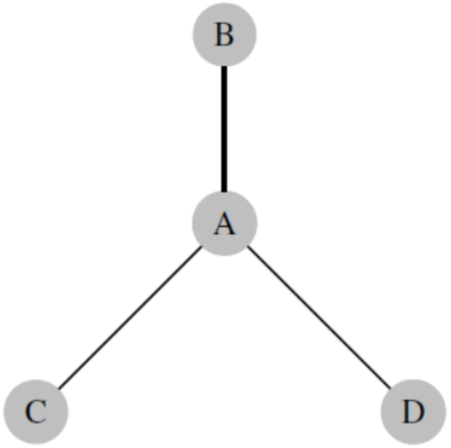

Fig. 2. Basic graphs of LAG method. Node A is a junction and nodes B,C and D are paths.

Digital computers deviated the way of template matching from optical to logical ones. Peephole method was one of the earliest logical approaches to digital template matching. Template matching technique evolved and offered using moments [1] and Fourier series later [3]. The former method made an invaluable translation-rotation-scale invariant a.k.a TRS, a tool for template matching [4]. Another set of commonplace TRS moments in the vicinity of pattern recognition are Zernike [5] and Fourier-Mellin moments that implemented in character recognition as well [6, 7].

Sakoe offered using dynamic programming (DP) technique to template matching problem [8]. Dynamic time warping (DTW) used to find the minimum distance between a given template and its corresponding candidate [8, 9]. DP-based template matching techniques make the comparison of two non-equal length vectors possible. Therefore, character matching would be possible while aspect ratio remains intact through DP.

Template matching practice, however, remained limited to printed character recognition systems [10]. To cover handwritten characters, another useful recognition method called structural analysis, introduced [1]. Freeman introduced a novel method on encoding curvatures in [11]. The proposed method, currently called "chain codes", was proposed in order to be used in image compression however, later it found practice in character recognition efforts [10]. Chain codes were developed in encoding efforts and a more sophisticated one is presented in [12]. Chain codes are founded on quantization process and decode a curvature to a line with a certain slope. As depicted in figure 1, 4-connected and 8-connected grids are two types of commonplace chain codes being used. The higher slopes in a grid, the lower quantization error in the coded curvature. Pavlidis at Bell laboratories proposed a thinning method suited for multi-font document recognition system [13]. The before said method uses line adjacency graphs (LAGs) which tries to stack semi-linear group of segments. Pavlidis addresses two types of graph nodes as junction and paths as shown in figure 2.

Structure analysis method alleviated template matching defects in handwritten character recognition systems. The candidate with the lowest distance with the template considers to be the best match. Since characters are not connected, registration problem, however, is straightforward to deal with.

In a wider sense, trainable scoring techniques are amazingly suited to human activities like speech and handwriting recognition [14]. Hidden markov models a.k.a HMMs have long been used in speech recognition systems. Due to probabilistic features of human writing systems, HMM has also been used to serve character recognition purposes [15]. Neural networks are another useful matching asset in character recognition systems [16,17,18,19]. Neural networks are trainable and their usage is less complicated as HMMs while multi layer perceptrons (MLPs) need more training data compared with HMMs [20]. Despite HMMs' more accurate performance in [20], MLPs show a better performance in speech recognition activities [21].

Right-to-left, cursive and connected scripts, however, are the most challenging scripts in recognition systems [22]. Arabic and Farsi scripts are two well-known cursive scripts widely using. Reading these scripts requires an excessive segmentation process, say, word segmentation [23, 24]. Technically speaking, a Farsi and/or Arabic word consists of connected characters (see figure 3). Therefore, a vital step in reading Farsi/Arabic characters is to rightly segment all of the characters.

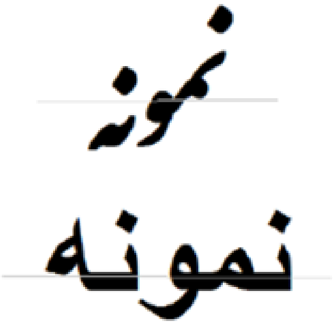

Fig. 3. Two samples of Farsi word “Sample”. This word consists of 5 characters N-M-U-N-H. It is noteworthy that unlike Arabic, vowels are not being written in formal Farsi handwriting.

This paper is organized as follows. In the section 2 the proposed framework is shortly introduced. In section 3, the preprocessing methods are discussed. Section 4 is devoted to segmentation procedures. Section 5 pays attention to training process and scoring. Finally, section 6 reveals the results of our proposed method.

2. DISCUSSING THE PROBLEM

Reading and checking printed score sheets for a massive number students is a cumbersome task which is not achievable by human resources. Kanoon Farhangi Amoozesh is a test-conducting foundation in Iran holds weekly exams among 400,000 students. In order to analyze students' educational background, one needs to certify the student claimed GPA according to printed documents. An Iranian high-school score-sheet issued by ministry of education is shown in figure 4.

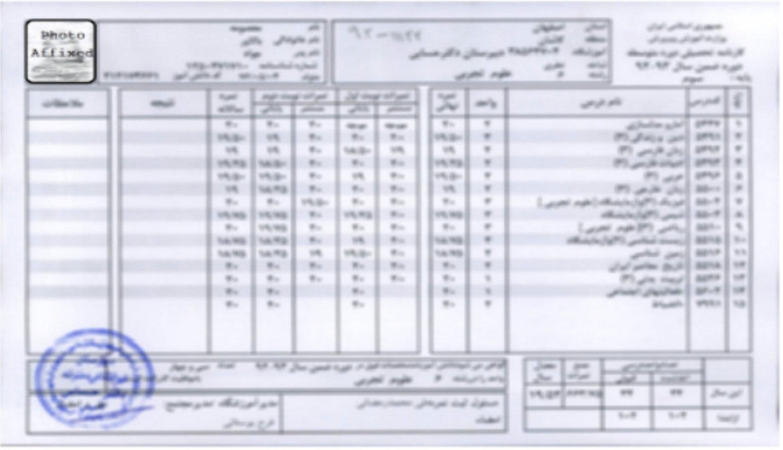
Photo Affixed

Fig. 4. A standard Iranian high-school score-sheet

3. PREPROCESSING

The devised system works along with a web-server in which server sends uploaded score-sheets to character recognition system and its output is fed back to the web-server (see figure 5).

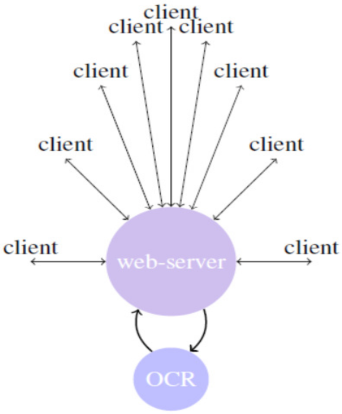

Fig. 5. Topology of the devised system

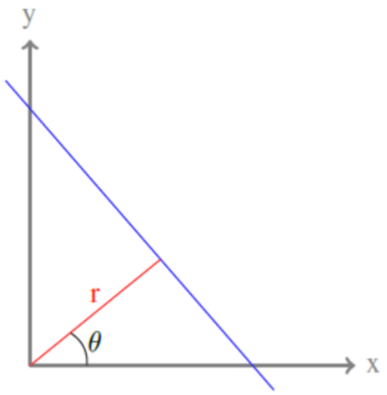

Fig. 6. Representation of a line in polar coordinates

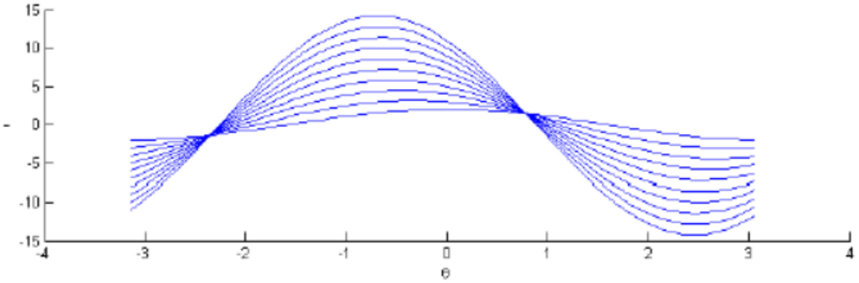

Fig.7. Sinusoid curves intersect on points located on a line. Curves are intersecting in two distinct points. It means that all of the points are located on a line.

A major problem with this case is image acquisition phase. As long as images are taken by clients, there is no control over the scanning procedure. The input images to the character recognition system are prone to rotation, scale manipulation and translation. Therefore, a preprocessing step is strictly required.

The most annoying scanning-mismatch is rotation. To remove rotation effects on reading characters, the perpendicular lines of the table are used. Hough line transform is used to extract lines and their angles [25]. As shown in figure [6], the Hough line transform first expresses a line in the polar system as below.

$$r = x\cos(\theta) + y\sin(\theta) \tag{1}$$

Each point P=(x_0,y_0) identifies a Sinusoidal curve. Points located on a straight line result in Sinusoidal curves that intersect in polar coordinates (see figure 7). In a real case, there are many points and many Sinusoidal curves in result. The points with the majority of intersections identify straight lines. In this venue one looks for the longest lines and computes their angle. After that, the input image will be rotated in inverse direction. Figure 8 depicts an instance of recovered lines using Hough method.

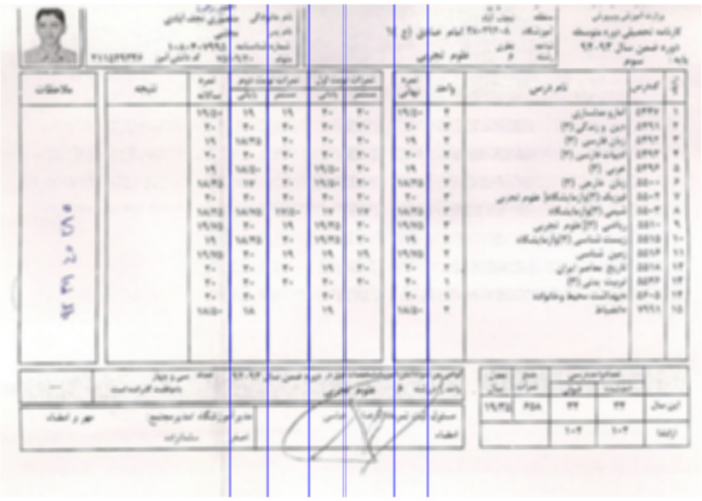

Fig. 8. A sample of rotation correction by Hough line transform. The rotation is removed by computing the rotation angle and inversely applying it to the image. In this method the horizontal lines are removed by filtering.

4. SEGMENTATION

A top-to-bottom raster scanning is required to find the position of lines whole the score-sheet. Before that, a canny edge detector should be applied to the image to remove redundant image data. In the case of characters, canny preserves the contours of the characters. Moreover, line height is obtainable in this way. Extracted lines are depicted in figure 9. This method is scale, translation and rotation invariant since it just attempts to find pixels concentration regions.

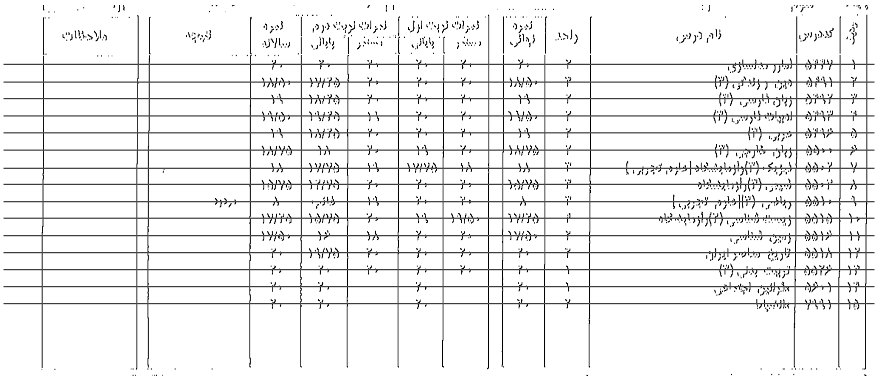

Fig. 9. Finding the rows of the score-sheets. The centers of the rows are emphasized by solid horizontal lines. The image is cropped to safeguard student's personal information.

In a similar manner, the table columns can be resolved easily. In the case the rows are not detectable, the input image would be rejected.

The last step to be paved is character segmentation. The output of row detection is fed to character parser. Character parser extracts a block containing the character. Fortunately, numerical digits are not connected in Farsi handwriting system. However, the character "point" or

"." is troublesome in Farsi. The mentioned character prints like "slash" or "/" in both Farsi and Arabic writing systems. The mentioned character sometimes connects two numerical digits specifically when scanning resolution is not fair enough. Figure 10 shows the output of character parser. To deal with the problem of parsing connected characters like figure 10.b, the authors proposed the average character width method in which the width of characters obtains for every row and the blocks wider than twice the width of average width considered as connected blocks. Average width method is life-savor in the case of ink spattering as well.

5. SCORING

The proposed multi-font character recognition system should be flexible about font change. Moreover, it should cover handwritten numerical digits. To cover this wide range of formation changes, artificial neural networks have been utilized [26]. In the proposed method multilayer perceptrons are used and back-propagation (BP) training method adjusts the weights of the perceptrons. Figure 11 illustrates an example of neural network with 4 hidden layers. Hidden layers are not accessible and their weight cannot be easily changed. Back-propagation method tries to minimize the distance between the real output, y_k, and the desired output, d_k. This distance can be formulated as below:

$$\epsilon = \frac{1}{2}\sum_{k=1} N(d_k - y_k)^2 \tag{2}$$

Where N is the number of neurons. The effect of neurons' weights on the output error is represented by gradient of ε_k.

$$\Delta\epsilon_k = \frac{\partial\epsilon}{\partial w_{kj}} \tag{3}$$

Using the steepest descent gradient algorithm [27] it leads to:

$$w_{kj}(m+1) = w_{kj}(m) + \Delta w_{kj}(m) \tag{4}$$

and w_{kj} is as below:

$$\Delta w_{kj} = -\eta\frac{\partial\epsilon}{\partial w_{kj}} \tag{5}$$

η is coined as learning rate. The output of each perceptron, y_k, is the weighted sum of previous layer perceptrons. In other words:

$$z_k = \sum_j w_{kj}x_j \tag{6}$$

z_k applies to sigmiod function :

$$F_N(x) = (1 + e^{-x})^{-1}$$

Wherein:

$$y_k = F_N(z_k) \tag{7}$$

Exploiting the sigmoid function results in:

$$\frac{\partial\epsilon}{\partial w_{kj}} = \frac{\partial\epsilon}{\partial z_k}\frac{\partial z_k}{\partial w_{kj}} \tag{8}$$

Then using (6) and the fact that $y_j(p - 1)=x_j(p)$ leads to:

$$\frac{\partial \epsilon}{\partial w_{kj}} = x_j(p) = y_j(p-1) \tag{9}$$

where p is the output layer. Let define a new parameter ɸ as follows:

$$\Phi_k(p) = -\frac{\partial \epsilon}{\partial z_k(p)} \tag{10}$$

Using this new parameter and obtain:

$$\frac{\partial \epsilon}{\partial w_{kj}} = -\Phi_k(p)x_j(p) = -\Phi_k y_j(p-1) \tag{11}$$

Doing some math and to get:

$$\Delta w_{kj} = \eta \phi_k(p) x_j(p) = \eta \Phi_k(p) y_j(p-1) \tag{12}$$

Invoking the chain rule in equation (10) results in:

$$\Phi_k = -\frac{\partial \epsilon}{\partial z_k} = -\frac{\partial \epsilon}{\partial y_k}\frac{\partial y_k}{\partial z_k} \tag{13}$$

The latter equation is used for the output layer. Differentiating from (2) results in:

$$\frac{\partial \epsilon}{\partial y_k} = -(d_k - y_k) = y_k - d_k \tag{14}$$

Exploiting the sigmoid property which is

$$y_k = F_N(z_k) = y_k(1 - y_k) \tag{15}$$

Therefore

$$\Phi_k = y_k(1 - y_k)(d_k - y_k) \tag{16}$$

Invoking (5) and (6) to obtain Δw_{kj} as below:

$$\Delta w_{kj}(p) = \eta \Phi_k(p) y_j(p-1) \tag{17}$$

The Δw_{ji} for the hidden layers are being computed as follows.

$$\Delta w_{ji} = -\eta \frac{\partial \epsilon}{\partial z_j} y_j(r-1) = \eta \Phi_j(r) y_i(r-1) \tag{18}$$

In order to update $ɸ_j$ for the hidden layers, the following equation is being used.

$$\begin{aligned}\Phi_j(r) = \frac{\partial y_j}{\partial z_j} \sum_k \Phi_k(r+1) w_{kj}(r+1) = \\ y_j(r)[1 - y_j(r)] \sum_k \Phi_k(r+1) w_{kj}(r+1)\end{aligned} \tag{19}$$

Back-propagation method converges more quickly than Adaline and Madaline methods [26].
In order to train the neural network, the segmented digit obtained from the segmentation process feeds to neural network as a 20x20 binary image. The training set consists of 42 classes without rejection class. Each class consists of 300 samples of 300 Farsi/Arabic fonts.

6. RESULTS

The proposed character recognition approach is implemented using C language along with openCV library. It is able to read score-sheets and report the results in both plain text and HTML format. The utilization is as simple as follows:

```
$ ocr -i < path to input file > [-h]
```

Wherein "-h" comes if the HTML report is needed. The system reads every 1200x800 pixel score-sheet in less than 2 seconds on a core i5 2.4GHz personal computer. The devised system brings a high degree of precision. In average, the presented system correctly reads every numerical characters in 0.015 seconds with an error rate less than 2%. The accuracy of the Google's Tesseract is less than 40% for similar tests.

ACKNOWLEDGEMENTS

Writing and editing this paper was not possible without invaluable help of Omid Sefat, Ali Golestan and Farhad Baybordi in Kanoon Farhangi Amoozesh. Moreover, the obstacles we jumped over were shortened by Kazem Ghalamchi.

REFERENCES

[1] S.Mori, C.Suen, and K. Yamamoto, “Historical review of ocr research and development,” Proceedings of the IEEE, vol. 80, pp. 1029–1058, Jul 1992.

[2] Y.Hongo and Y.Nitta, “Pattern recognition apparatus,” Dec. 9 1986. US Patent 4,628,533.

[3] C.T.Zahn and R.Z. Roskies, “Fourier descriptors for plane closed curves,” Computers, IEEE Transactions on, vol. C-21, pp. 269–281, March 1972.

[4] M.-K. Hu, “Visual pattern recognition by moment invariants,” Information Theory, IRE Transactions on, vol. 8, pp. 179–187, February 1962.

[5] D.Xiao and L.Yang, “Gait recognition using zernike moments and bp neural network,” in Networking, Sensing and Control, 2008. ICNSC 2008. IEEE International Conference on, pp. 418–423, April 2008.

[6] C.Kan and M.D.Srinath, “Invariant character recognition with zernike and orthogonal fouriermellin moments,” Pattern Recognition, vol. 35, no. 1, pp. 143 – 154, 2002. Shape representation and similarity for image databases.

[7] L.Torres-Mendez, J.Ruiz-Suarez, L.Sucar, and G.Gomez, “Translation, rotation, and scale-invariant object recognition,” Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 30, pp. 125–130, Feb 2000.

[8] H.Sakoe and S.Chiba, “Dynamic programming algorithm optimization for spoken word recognition,” Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 26, pp. 43–49, Feb 1978.

[9] D.Falahati, M.Helforoush, H.Danyali, and M.Rashidpour, “Static signature verification for farsi and arabic signatures using dynamic time warping,” in Electrical Engineering (ICEE), 2011 19th Iranian Conference on, pp. 1–1, May 2011.

[10] A.Sinha, “An improved recognition module for the identification of handwritten digits,” 1999.

[11] H.Freeman, “On the encoding of arbitrary geometric configurations,” Electronic Computers, IRE Transactions on, vol. EC-10, pp. 260–268, June 1961.

[12] G.Schuster and A.Katsaggelos, “An optimal polygonal boundary encoding scheme in the rate distortion sense,” Image Processing, IEEE Transactions on, vol. 7, pp. 13–26, Jan 1998.

[13] T.Pavlidis, “A vectorizer and feature extractor for document recognition,” Computer. Vision Graph. Image Process., vol. 35, pp. 111–127, July 1986.

[14] L.Rabiner, “A tutorial on hidden markov models and selected applications in speech recognition,” Proceedings of the IEEE, vol. 77, pp. 257–286, Feb 1989.

[15] O.E. Agazzi and S. shiaw Kuo, “Hidden markov model based optical character recognition in the presence of deterministic transformations,” Pattern Recognition, vol. 26, no. 12, pp. 1813 – 1826, 1993.

[16] E.Alpaydin, “Optical character recognition using artificial neural networks,” in Artificial Neural Networks, 1989., First IEE International Conference on (Conf. Publ. No. 313), pp. 191–195, Oct 1989.

[17] R.Arnold and P.Miklos, “Character recognition using neural networks,” in Computational Intelligence and Informatics (CINTI), 2010 11th International Symposium on, pp. 311–314, Nov 2010.

[18] F.Yang and F.Yang, “Character recognition using parallel bp neural network,” in Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on, pp. 1595–1599, July 2008.

[19] A.Gupta, M.Srivastava, and C.Mahanta, “Offline handwritten character recognition using neural network,” in Computer Applications and Industrial Electronics (ICCAIE), 2011 IEEE International Conference on, pp. 102–107, Dec 2011.

[20] E.Hatzipantelis, A.Murray, and J.Penman, “Comparing hidden markov models with artificial neural network architectures for condition monitoring applications,” in Artificial Neural Networks, 1995., Fourth International Conference on, pp. 369–374, Jun 1995.

[21] A.Waibel, T.Hanazawa, G.Hinton, K.Shikano, and K. Lang, “Phoneme recognition: neural networks vs. hidden markov models vs. hidden markov models,” in Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on, pp. 107–110 vol.1, Apr 1988.

[22] A.Cheung, M.Bennamoun, and N. Bergmann, “A recognition-based arabic optical character recognition system,” in Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on, vol. 5, pp. 4189–4194 vol.5, Oct 1998.

[23] V.Margner and M.Pechwitz, “Synthetic data for arabic ocr system development,” in Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on, pp. 1159–1163, 2001.

[24] R.Prasad, S.Saleem, M.Kamali, R.Meermeier, and P.Natarajan, “Improvements in hidden markov model based arabic ocr,” in Pattern Recognition, 2008. ICPR 2008. 19th International Conference on, pp. 1–4, Dec 2008.

[25] G.Stockman and L.G.Shapiro, Computer Vision. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1st ed., 2001.

[26] D.Graupe, Principles of Artificial Neural Networks. River Edge, NJ, USA: World Scientific Publishing Co., Inc., 1997.

[27] Z.Tian-liang, “Solving non-linear equation based on steepest descent method,” in Information and Computing (ICIC).