World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:12, No:12, 2018

Arabic Character Recognition Using Regression Curves with the Expectation Maximization Algorithm

Abdullah A. AlShaher

***Abstract*—**In this paper, we demonstrate how regression curves can be used to recognize 2D non-rigid handwritten shapes. Each shape is represented by a set of non-overlapping uniformly distributed landmarks. The underlying models utilize 2nd order of polynomials to model shapes within a training set. To estimate the regression models, we need to extract the required coefficients which describe the variations for a set of shape class. Hence, a least square method is used to estimate such modes. We then proceed by training these coefficients using the apparatus Expectation Maximization algorithm. Recognition is carried out by finding the least error landmarks displacement with respect to the model curves. Handwritten isolated Arabic characters are used to evaluate our approach.

***Keywords*—**Shape recognition, Arabic handwritten characters, regression curves, expectation maximization algorithm.

I. Introduction

SHAPE recognition has been the focus of many researchers since seven past decades [1] and attracted many communities in the field of pattern recognition [2], artificial intelligence[3], signal processing [4], image analysis [5], and computer vision [6]. The difficulties arise when the shape under study exhibits high degree in shape variation: as in handwritten characters [7], digits [8], face detection [9], and gesture authentication [10]. For a single data, shape variation is limited and cannot be captured ultimately due to the fact that single data does not provide sufficient information and knowledge about the data; therefore, multiple existence of data provides better understanding of shape analysis and manifested by mixture models [11]. Because of the existence of multivariate data under study, there is always the requirement to estimate the parameters that describe the data that is encapsulated within a mixture of shapes.

The literature demonstrates many statistical and structural approaches with various algorithms to model shape variations using supervised and unsupervised learning [12] algorithms. In precise, the powerful Expectation Maximization algorithm of Dempster [13] comes that has been used widely for such cases. The EM algorithm revolves around two step procedure. The expectation E step revolves around estimating the parameters of a log-likelihood function and pass it to the Maximization M step. In a maximization (M) step, the algorithm computes parameters maximizing the expected log-likelihood found on the E step. The process is iterative one until all parameters comes to unchanged. For instance, Jojic and Frey [14] have used the EM algorithm to fit mixture models to the appearance manifolds for faces. Bishop and Winn [15] have used a mixture of principal components analyzers to learn and synthesize variations in facial appearance. Vasconcelos and Lippman [16] have used the EM algorithm to learn queries for content-based image retrieval. Finally, several authors have used the EM algorithm to track multiple moving objects [17]. Revov et al. [18] have developed a generative model which can be used for handwritten character recognition. Their method employs the EM algorithm to model the distribution of sample points.

Curves are used widely by research in the computer vision society [1]-[5]. Curvatures are mainly used to distinguish different shapes such as characters [6], digits, faces [2], and topographic maps [3]. Curve fitting [18], [19] is the process of constructing a 2nd order or higher mathematical function that has the best fit to a series of landmark points. A related topic is regression analysis that stresses on probabilistic conclusion on how uncertainty is present when fitting a cure to a set of data landmarks with marginal errors. Regression curves are applied for data visualization [12], [13] to capture the values of a function with missing data [14] and to gain the relationship of multiple variables.

In this paper, we demonstrate how curves are used to recognize 2D handwritten shapes by fitting 2nd order of polynomial quadratic function to a set of landmarks points presented in a shape. We then train such curves to capture the optimal characteristics of the shapes in the training sets of curves. Handwritten Arabic characters are used and tested in this investigation.

II. Regression Curves

We would like to extract the best fit modes that describe the shapes under study, hence, a multiple image shapes are required and is explained by a training sets of class shape ω and the complete sets of shape classes denoted by Ω. Let us assume that each training set is represented by the following 2D training patterns as a long vector

$$X^{\omega} = ((x_1^{\omega_1}, y_1^{\omega_1}), .., (x_k^{\omega_1}, y_k^{\omega_1}), (x_2^{\omega_2}, y_2^{\omega_2}), .., (x_k^{\omega_2}, y_k^{\omega_2}), (x_1^{\omega_T}, y_1^{\omega_T}), .., (x_k^{\omega_T}, y_k^{\omega_T})) \quad (1)$$

Our model here is a polynomial of higher order. In this case, we choose 2nd order of quadratic curves. Consider the following generic form of a polynomial of order *j*

Abdullah A. AlShaher is with the Department of Computers and Information Systems, College of Business Studies, The Public Authority for Applied Education and Training, Ardiyah, The State of Kuwait (e-mail: aa.alshaher@paaet.edu.kw).

$$f(x_k^{\omega_T}) = a_0 + a_1 x_k^{\omega_T} + a_2 (x_k^{\omega_T})^2 + a_3 (x_k^{\omega_T})^3 + \cdots + a_j (x_k^{\omega_T})^j = a_0 + \sum_{\tau=1}^{j} a_x (x_k^{\omega_T})^{\tau} \quad (2)$$

The nonlinear regression above requires the estimation of the coefficients that best fit the sample shape landmarks, we approach the least square error between the data y and f(x) in

$$err = \sum (d_i)^2 = (y_1^{\omega_T} - f(x_1^{\omega_1}))^2 + (y_2^{\omega_T} - f(x_2^{\omega_1}))^2 + (y_3^{\omega_T} - f(x_3^{\omega_1}))^2 + (y_4^{\omega_T} - f(x_4^{\omega_1}))^2 \quad (3)$$

where the goal is to minimize the error, substituting the form of (11) to a general least square error

$$err = \sum_{k=1}^{T} (y_k^{\omega_T} - (a_0 + a_1 x_k^{\omega_T} + a_2 x_k^{\omega_T 2} + a_3 x_k^{\omega_T 3} + \cdots + a_j x_k^{\omega_T j}))^2 \quad (4)$$

where T is the number of pattern set, k is the current data landmarks point being summed, j is the order of polynomial equation. Rewriting (14) in a more readable format

$$err = \sum_{k=1}^{T} (y_k^{\omega_T} - (a_0 + \sum_{k=1}^{j} a_k x^k))^2 \quad (5)$$

Finding the best fit curve is equivalent to minimizing the squared distance between the curve and landmark points. The aim here is to find the coefficients, hence, solving the equations of taking the partial derivative with respect each coefficients a_0, a_k; for $k = 1..j$ and set each to zero in

$$\frac{\partial err}{\partial a_0} = \sum_{k=1}^{T} (y_k^{\omega_T} - (a_0 + \sum_{K=1}^{T} a_k x^k)) = 0 \quad (6)$$

$$\frac{\partial err}{\partial a_1} = \sum_{k=1}^{T} (y_k^{\omega_T} - (a_0 + \sum_{K=1}^{T} a_k x^k)) x_k^{\omega_T} = 0 \quad (7)$$

$$\frac{\partial err}{\partial a_2} = \sum_{k=1}^{T} (y_k^{\omega_T} - (a_0 + \sum_{K=1}^{T} a_k x^k)) x_k^{\omega_T 2} = 0 \quad (8)$$

Rewriting upper equations in the form of matrix and applying linear algebra matrix differentiation, we get

$$\begin{bmatrix} T & \sum_{k=1}^{T} x_k^{\omega_T} & \sum_{k=1}^{T} x_k^{\omega_T 2} \\ \sum_{k=1}^{T} x_k^{\omega_T} & \sum_{k=1}^{T} x_k^{\omega_T 2} & \sum_{k=1}^{T} x_k^{\omega_T 3} \\ \sum_{k=1}^{T} x_k^{\omega_T 2} & \sum_{k=1}^{T} x_k^{\omega_T 3} & \sum_{k=1}^{T} x_k^{\omega_T 4} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{T} y_k^{\omega_T} \\ \sum_{k=1}^{T} x_k^{\omega_T} y_k^{\omega_T} \\ \sum_{k=1}^{T} x_k^{\omega_T 2} y_k^{\omega_T} \end{bmatrix} \quad (9)$$

Choosing Gaussian elimination procedure to rewrite the upper equation in more solvable in

$$Ax = B \quad (10)$$

where

$$\mathrm{A} = \begin{bmatrix} T & \sum_{k=1}^{T} x_k^{\omega_T} & \sum_{k=1}^{T} x_k^{\omega_T 2} \\ \sum_{k=1}^{T} x_k^{\omega_T} & \sum_{k=1}^{T} x_k^{\omega_T 2} & \sum_{k=1}^{T} x_k^{\omega_T 3} \\ \sum_{k=1}^{T} x_k^{\omega_T 2} & \sum_{k=1}^{T} x_k^{\omega_T 3} & \sum_{k=1}^{T} x_k^{\omega_T 4} \end{bmatrix}, \mathrm{X} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}, \mathrm{B} = \begin{bmatrix} \sum_{k=1}^{T} y_k^{\omega_T} \\ \sum_{k=1}^{T} x_k^{\omega_T} y_k^{\omega_T} \\ \sum_{k=1}^{T} x_k^{\omega_T 2} y_k^{\omega_T} \end{bmatrix} \quad (11)$$

solving for X to find the coefficients A, B in

$$X = A^{-1} * B \quad (12)$$

The outcome would be the coefficients a_0, a_1, a_2. We follow the similar procedure to find the remaining coefficients of the landmarks points.

III. Learning Regression Curves

It has been known that when using learning algorithms to train models of such a case, the outcome is trained models with superior performance than those of untrained models bishop [19]. In this stage, we are concerned by capturing the optimal curve coefficients which describe the patterns variations under testing; hence, training is required by fitting the Gaussian mixtures of curve coefficient models to set of shape curve patterns. The previous approaches regard producing variations in shapes that of a linear fashion. To produce more complex shape variations, we have to proceed by employing non-linear deformation of a set of curve coefficients. Unsupervised learning is used encapsulated within a framework of the apparatus Expectation Maximization EM algorithm. The idea is borrowed from Cootes [20] of constructing point distribution models; however, the algorithm is transformed to learn regression curves coefficients α_t similar to that approach of AlShaher [21]. Suppose that a set of curve coefficients α_t for a set of training patterns is $t = (1 \ldots T)$ where T is the complete set of training curves is represented in a long vector of coefficients :

$$\alpha_t = (a_{t_{1_1}}, a_{t_{1_2}}, a_{t_{1_3}}, a_{t_{2_1}}, a_{t_{2_2}}, a_{t_{2_3}}, \ldots, a_{t_{in}} a_{t_{in}}) \quad (13)$$

The mean vector of coefficient patterns is represented by

$$\mu = \frac{1}{T} \sum_{t=1}^{T} \alpha_t \quad (14)$$

The covariance matrix is then constructed by

$$\Sigma = \frac{1}{T} \sum_{t=1}^{T} (\alpha_t - \mu)(\alpha_t - \mu)^T \quad (15)$$

The following approach is based on fitting a Gaussian mixture models to the set of training examples of curve coefficients. We further assume that training patterns are independent from one each other; thus, they are neither flagged nor labelled to any curve class. Each curve class ω belongs to the set of curve classes Ω has its own mean μ and covariance matrix ∑. With these ingredients, we establish the likelihood function for the set of the curve patterns in

$$p(\alpha_t) = \prod_{t=1}^{T} \sum_{w=1}^{\Omega} p(\alpha_t | \mu_w, \Sigma_w) \quad (16)$$

where the term $p(\alpha_t | \mu_w, \Sigma_w)$ is the probability for drawing curve pattern α_t from the curve-class ω. Associating the above likelihood function with the Expectation Maximization algorithm, the likelihood function can be written to be iterative process of two steps. The process revolves around estimating the expected log-likelihood function iteratively in

$$q_L\left(C^{(n+1)}\middle|C^{(n)}\right) = \sum_{t=1}^{T} \sum_{w=1}^{\Omega} P(\alpha_t, \mu_w^{(n)}, \Sigma_w^{(n)})\ X \ln p(\alpha_t|\mu_w^{(n+1)}, \Sigma_w^{(n+1)}) \quad (17)$$

where the quantity and $\mu_w^{(n)}$ and $\sum_w^{(n)}$ are the estimated mean curve vector and variance covariance matrix both at iteration (n) of the algorithm. The quantity $p(\alpha_t, \mu_w^{(n)}, \Sigma_w^{(n)})$ is the *a posteriori* probability that the training pattern curve belong to the curve-class ω at iteration n of the algorithm. The term $p(\alpha_t|\mu_w^{(n+1)}, \Sigma_w^{(n+1)})$ is the probability of distribution of curve-pattern α_t belonging to curve-class ω at iteration (n + 1) of the algorithm; thus, the probability density to associate curve-patterns α_t for (t = 1 … T) to class curve-class ω are estimated by the updated construction of the mean-vector $\mu_w^{(n+1)}$, and covariance matrix $\sum_w^{(n+1)}$ at iteration n+1 of the algorithm. According to the EM algorithm, it revolves around estimation the expected log-likelihood function within two iterative processes. In the M or maximization step of the algorithm, our aim is to maximize the curve mean-vector $\mu_w^{(n+1)}$, and covariance matrix $\sum_w^{(n+1)}$, while, in the E or expectation step, the aim is to estimate the distribution of curve-patterns at iteration n along with the mixing proportion parameters for curve-class ω.

In the *E,* or Expectation step of the algorithm, the a posteriori curve-class probability is updated by applying the Bayes factorization rule to the curve-class distribution density at iteration n+1. The new estimate is computed by

$$p\left(\propto_t, \mu_w^{(n)}, \Sigma_w^{(n)}\right) = \frac{p(\alpha_t|\mu_w^{(n)}, \Sigma_w^{(n)})\ \pi_w^{(n)}}{\sum_{w=1}^{\Omega} p(\alpha_t|\mu_w^{(n)}, \Sigma_w^{(n)})\ \pi_w^{(n)}} \quad (18)$$

where the revised curve-class ω mixing proportions $\pi_w^{(n+1)}$ at iteration *(n + 1)* is computed in

$$\pi_w^{(n+1)} = \frac{1}{T} \sum_{t=1}^{T} p(\alpha_t|\mu_w^{(n)}, \Sigma_w^{(n)}) \quad (19)$$

With that at hand, the distributed curve-pattern α_t to the class-curve ω is Gaussian distribution and is classified according to

$$p\left(\alpha_t\middle|\mu_w^{(n)}, \Sigma_w^{(n)}\right) = \frac{1}{(2\pi)^L \sqrt{|\Sigma_w^{(n)}|}} \exp\left[-\frac{1}{2}\left(\alpha_t - \mu_w^{(n)}\right)^T X \left(\Sigma_w^{(n)}\right)^{-1} X\ (\alpha_t - \mu_w^{(n)})\right] \quad (20)$$

In the *M*, or Maximization step, our aim is to maximize the curve-class ω parameters. The updated curve mean-vector $\mu_w^{(n+1)}$ estimate is computed by

$$\mu_w^{(n+1)} = \sum_{t=1}^{T} p(\alpha_t, \mu_w^{(n)}, \Sigma_w^{(n)})\alpha_t \quad (21)$$

And the new estimate of the curve-class covariance matrix is weighted by

$$\Sigma_w^{(n+1)} = \sum_{t=1}^{T} p\left(\alpha_t, \mu_w^{(n)}, \Sigma_w^{(n)}\right) X(\alpha_t - \mu_w^{(n)}) \left(\alpha_t - \mu_w^{(n)}\right)^T \quad (22)$$

Both E, and M steps are iteratively converged, the outcome of the learning stage is a set of curve-class ω parameters such as $\mu_w^{(n)}$ *and* $\sum_w^{(n)}$, hence the complete set of all curve-class **Ω** are computed and ready to be used for recognition.

IV. Recognition

In this stage, we focus of utilizing the parameters extracted from the learning phase for the purpose of shape recognition. Here, we assume that the testing shapes

$$f(t) = \sum_{t=1}^{X} \sum_{i=1}^{n} (x_{t_i}, y_{t_i}), where\ (i = 1..n), (t = 1..X) \quad (23)$$

Hence, each testing pattern is represented by

$$\chi_t = \left((x_{t_1}, y_{t_1}), (x_{t_2}, y_{t_2}), \ldots (x_{t_i}, y_{t_i}) \right) \text{ for } (t = 1 .. X) \quad (24)$$

Such testing patterns are classified based on the computing the new point position of testing data χ after projecting the sequence of curve-coefficients to the testing data in

$$f(x, y) = \sum_{i=1}^{n} (\chi_{t_{y_i}} - (\alpha_{t_{i1}} \chi_{t_{x_i}}^2 + \alpha_{t_{i2}} \chi_{t_{x_i}} + \alpha_{t_{i3}})) \quad (25)$$

So, the sample shape χ_t is registered to class ω which has the highest probability using Bayes rule over the total curve-classes **Ω** in

$$\arg\min \frac{f(x,y)}{\sum_{w=1}^{\Omega} f(x,y)} \quad (26)$$

V. Experiments

We have evaluated our approach with sets of Arabic handwritten characters. Here, we have used 23 shape-classes for different writers, each with 80 training patterns. In total, we have tested the approach with 1840 handwritten Arabic character shape patterns for testing and 4600 patterns for testing phase. Figs. 1 and 2 show some training patterns used in this paper. Fig. 3 shows single shapes and their landmarks representation.

Fig. 4 demonstrates regression sample curve-classes as a result of the training stage. Fig. 5 demonstrate the curve-classes **Ω** convergence rate graph as a function per iteration no. is the training phase. The graphs show how associated distributed probabilities for the set of curve-classes **Ω** converged in a few iterations.

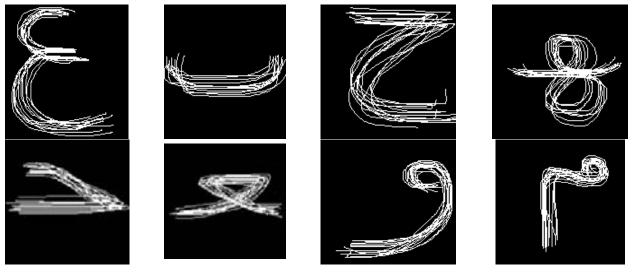

Fig. 1 Training sets sample

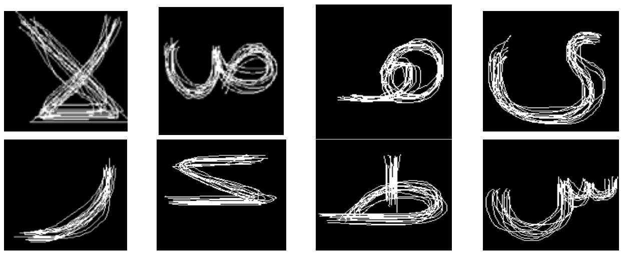

Fig. 2 Training set sample 2

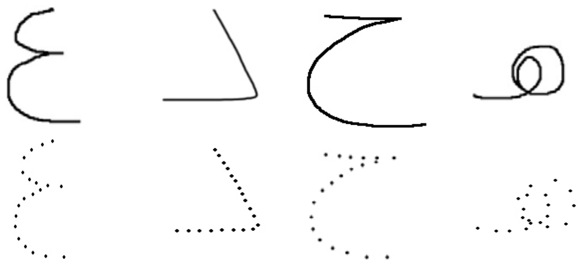

Fig. 3 Training patterns and extracted landmarks

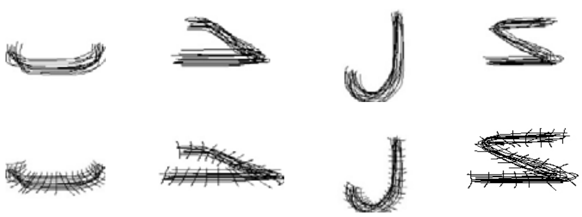

Fig. 4 Sample visual of regression Curve-classes

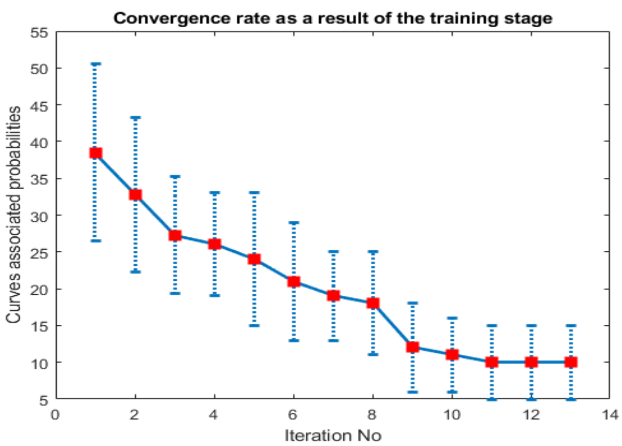

Fig. 5 Convergence rate as a function per iteration no.

To take the investigation further, we demonstrate how well the approach behaves in the presence of noise. In figure 6, we show how recognition rate is achieved when point position displacement error is applied. Test shape coordinates are being moved away from their original position. The figure shows the recognition rate fails to register shapes to their correct class in a few iterations and it fails completely when coordinates are moved away, yet, increasing variance significantly.

TABLE I
RECOGNITION RATE FOR SAMPLE SHAPES

Shape Name	Sample Shape	Test Size	Correct	False	Recognition Rate
Ain_1		200	191	9	95.5%
Baa		200	193	7	96.5%
Dal		200	194	6	97%
Faa		200	187	13	93.5%
Haa_1		200	196	4	98%
Ttah		200	180	20	90%
Hhah_1		200	178	22	89%
Ain_2		200	190	10	95%
Meem		200	182	18	91%
Seen		200	175	25	87.5%
Yaa		200	189	11	94.5%
Haa_2		200	193	7	96.5%
Waw		200	183	17	91.5%
Ain_2		200	195	5	97.5%
Hhah_2		200	172	28	86%
Kaf_1		200	196	4	98%
Lam_1		200	193	7	96.5%
Lam_2		200	181	19	90.5%
Raa		200	196	4	98%
Ssad		200	176	24	88%
Kaf_2		200	192	8	96%
Noon		200	196	4	98%
Total	22 classes	4400	4400	6.18%	93.82%

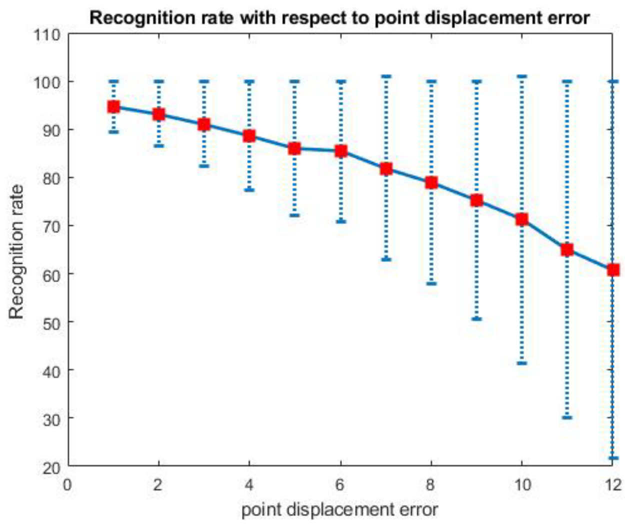

Fig. 6 Recognition rate as a function per iteration no with point position error

Table I shows recognition rates per curve-classes ω. The Table demonstrates recognition rates per curve-class. In total, we have achieved 94% recognition rate for such approach.

VI. CONCLUSION

In this paper, we have showed how Regression Curves can be used to model the variation of Handwritten Arabic characters. The 2nd order of Polynomials curves were injected along the skeleton of the proposed shape under study, where the appropriate coefficients which describe the shape were extracted. We then have used the Apparatus of the Expectation Maximization Algorithm to train the set of extracted coefficients within a probabilistic framework to capture the optimal shape variations coefficients. The set of best fitted parameters were then projected in order to recognize handwritten shapes using Bayes rule of factorization. The proposed approach has been evaluated on sets of Handwritten Arabic Shapes for multiple different writers that we have achieved a recognition rate of nearly 94% on corrected registered shape classes.

VII. Future Work

In this research, there are some shortcomings to the approach. One of which is that the is missing thoroughly comparison between the proposed method and conventional method to demonstrate the effectiveness of such complicated method. The second is that the extracted parameters from the training stage have not been utilized for the purpose of recognition stage in a statistical framework. Thirdly, we further investigate how we use deformation parameters to fit the align curves to such sample shape, specifically when noise is present.

References

[1] J. Wood, "Invariant pattern recognition: A review," Pattern Recognition, vol. 29, no. 1, 1996, 1-17.
[2] Anil, K. Jain,Robert P. W. Duin, and Jianchang Mao: “Statistical Pattern Recognition: A Review”, IEEE Pattern Analysis and Machine Intelligence, vol 22, No. 1, PP 4-37, 2000.
[3] P. F. Baldi and K. Hornik, "Learning in linear neural networks: A survey," IEEE Transactions on Neural Networks, vol. 6, no. 4, 1995, 837-858.
[4] T. Y. Kong and A. Rosenfeld, "Digital topology: introduction and survey," Computer Vision, Graphics, and Image Processing, vol. 48, no. 3, pp. 357-393, 1989.
[5] T. R. Reed and J. M. H. Dubuf, "A review of recent texture segmentation and feature extraction techniques," CVGIP - Image Understanding, vol. 57, no. 3, pp. 359-372, 1993.
[6] S. Sarkar and K. L. Boyer, "Perceptual organization in computer vision - a review and a proposal for a classifactory structure", IEEE Transactions on Systems Man and Cybernetics, vol. 23, no. 2, 1993, 382-399.
[7] Lorigo L. M., Govindaraju V., “Offline Arabic Handwriting Recognition: A Survey”, IEEE Trans Pattern Anal Mach Intelligence, Vol: 28(5): PP 712-24, 2006.
[8] Ishani Patel, Virag Jagtap, Ompriya Kale, “A Survey on Feature Extraction Methods for Handwritten Digits Recognition”, International Journal of Computer Applications, Vol 107, No 12, PP 11-17, 2014.
[9] Muhammad Sharif, Farah Naz, Mussarat Yasmin, Muhammad Alyas Shahid, Amjad Rehman, “Face Recognition: A Survey”, Journal of Engineering Science and Technology Review, Vol 10, No 2, PP 166-177, 2017.
[10] Rafiqul Zaman Khan, Noor Adnan Ibraheem, “Hand Gesture Recognition: A Literature Review “International Journal of Artificial Intelligence & Applications, Vol.3, No.4, pp 161-174, 2012.
[11] Bruce George Lindsay, Mary L. Lesperance , “A review of semiparametric mixture models”, Journal of Statistical Planning and Inference, Vol: 47 , No: 1 PP 29-39 , 1995.
[12] Christopher M. Bishop, “Neural Networks for Pattern Recognition”, Clarendon Press, 1995, ISSN 0198538642.
[13] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the em algorithm, J. Roy. Statist. Soc. Ser. 39 (1977) 1–38.
[14] J. Brendan, N. Jojic, Estimating mixture models of images and inferring spatial transformations using the em algorithm, IEEE Comput. Vision Pattern Recognition 2 (1999) 416–422.
[15] C. Bishop, J. Winn, Non-linear Bayesian image modelling, Proceedings of Sixth European.
[16] N. Vasconcelos, A. Lippman, A probabilistic architecture for content-based image retrieval, Proceedings of International Conference on Computer Vision and Pattern Recognition, 2000, pp. 216–221.
[17] B. North, A. Blake, Using expectation-maximisation to learn dynamical models from visual data, Image Vision Computing. 17 (8) (1999) 611–616.
[18] M. Revow, C. Williams, G. E. Hinton, Using generative models for handwritten digit recognition, IEEE Trans. Pattern Anal. Mach. Intell. 20 (2) (1996) 592–606.
[19] Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer Science, and Business Media, 2006.
[20] T. Cootes, C. Taylor, A mixture model for representing shape variations. Image and Vision Computing, 17(1999) 403-409.
[21] Al Shaher Abdullah, Hancock Edwin, Learning mixtures of Point Distribution models with the EM algorithm. Pattern Recognition, 36(2003) 2805-2818.

Abdullah A. AlShaher has received his Bachelor degree in Computer Science from Huston Tillotson University Austin Texas, his Master degree in Computer Science from University of Denver, Denver Colorado, his Ph.D. in Computer Science form University of York UK. Currently taken the position of the head of Computer and Information Systems department in the Public Authority for Applied Education and Training. His main interests is Shape Recognition, Handwritten Character analysis, Optimization, Computer Vision, Pattern Recognition, and Signal Processing.