Journal of Computer Science 5 (3): 226-232, 2009
ISSN 1549-3636

Novel Moment Features Extraction for Recognizing Handwritten Arabic Letters

Gheith Abandah and Nasser Anssari
Department of Computer Engineering, University of Jordan, Amman 11942, Jordan

Abstract: Problem statement: Offline recognition of handwritten Arabic text awaits accurate recognition solutions. Most of the Arabic letters have secondary components that are important in recognizing these letters. However these components have large writing variations. We targeted enhancing the feature extraction stage in recognizing handwritten Arabic text. **Approach:** In this study, we proposed a novel feature extraction approach of handwritten Arabic letters. Pre-segmented letters were first partitioned into main body and secondary components. Then moment features were extracted from the whole letter as well as from the main body and the secondary components. Using multi-objective genetic algorithm, efficient feature subsets were selected. Finally, various feature subsets were evaluated according to their classification error using an SVM classifier. **Results:** The proposed approach improved the classification error in all cases studied. For example, the improvements of 20-feature subsets of normalized central moments and Zernike moments were 15 and 10%, respectively. **Conclusion/Recommendations:** Extracting and selecting statistical features from handwritten Arabic letters, their main bodies and their secondary components provided feature subsets that give higher recognition accuracies compared to the subsets of the whole letters alone.

Key words: Normalized central moments, Zernike moments, feature extraction, feature selection, handwritten Arabic letters

INTRODUCTION

Arabic letters are used in about 27 writing languages including Arabic, Persian, Kurdish, Urdu and Jawi[1]. Offline recognition of handwritten cursive text such as Arabic text is an active research problem[2,3]. Offline recognition of unconstrained handwritten cursive text must overcome many difficulties such as unlimited variation in human handwriting, similarities of distinct character shapes, character overlaps and interconnections of neighboring characters. Some progress has been made on recognizing handwritten Arabic text samples of limited vocabulary (e.g., IFN/ENIT database of handwritten Tunisian town names[4]). In ICDAR Arabic handwriting recognition competitions held in 2005 and 2007[5,6], best systems' accuracies improved from 76-87% on the IFN/ENIT database. However, recognition accuracy of unlimited vocabulary is still unacceptable for many applications.

In this study, we propose a new technique to extract statistical features of handwritten Arabic letters. We apply this technique in extracting moment features and show that this technique provides better feature sets that give higher recognition accuracies. This technique can be applied in extracting other state-of-the-art features such as chain code and gradient features[7].

Important features of Arabic writing: The Arabic alphabet has 28 basic letters[1,8]. Arabic is written from right to left and is always cursive. Each letter has multiple forms depending on its position in the word. Each letter is drawn in an isolated form when it is written alone and is drawn in three other forms when it is written connected to other letters in the word. For example, the letter Ain has four forms: isolated (ع), initial (عـ), medial (ـعـ) and final (ـع).

More than half the Arabic letters are composed of main body and secondary components. The secondary components are letter components that are disconnected from the main body. For example, Beh (ب) has a dot under its main body, Teh (ت) has two dots above its main body and Kaf (ك) has a zigzag enclosed within the main body.

The type and position of the secondary components are very important features of Arabic letters. For example, recognizing two dots below the main body are sufficient to recognize the letter Yeh (ي) because Yeh is the only letter that has two dots below its main body. Furthermore, some letters can only be distinguished by their secondary components. For example, Teh (ت) and Theh (ث) differ only by the number of dots above the main body and medial Teh (ـتـ) and medial Yeh (ـيـ) differ only by the position of the two dots.

Corresponding Author: Gheith Abandah, Department of Computer Engineering, University of Jordan, Amman 11942, Jordan

Table 1: Samples showing variations in drawing the secondary components

	1	2	3
A			
B			
C			
D			
E			
F			

There are important variations in drawing the secondary components; mostly in drawing two dots and three dots. As shown in Table 1, samples A1, A2 and A3, the two dots come in three variations: Two disconnected dots, two connected dots and horizontal dash. Samples B1, B2 and B3 show three variations in drawing the three dots: Three disconnected dots, one dot above horizontal dash and hat shape "^". Any secondary components classification process should take these variations into consideration[9].

One recognition difficulty is due to some writers' styles that replace the secondary components of isolated and final forms with main body curves. Table 1 shows some examples: Samples C1 and C2 show how the two dots of isolated Qaf are replaced, Samples D1 and D2 show how the one dot of isolated Noon is replaced and Samples E1 and E2 show how the zigzag of final Kaf is replaced.

Another difficulty in recognizing the secondary components comes when hasty writers draw them connected to the main body. For example, Sample E3 shows the zigzag connected to Kaf's body, Sample F1 shows the two dots connected to Teh's body, Sample F2 shows the three dots connected to Theh's body and Sample F3 shows the dot connected to Jeem's body.

Approach: In feature extraction, the whole image of the letter is normally used to extract statistical features such as moments. However, this approach does not exploit the full potential of the secondary components of the Arabic letters.

To exploit the potential of these components and to overcome the writing variations described above, we partition the letters into main body and secondary components. Then we extract features from the whole letter's image, from its main body and its secondary components.

This approach increases the number of extracted features by a factor of three, thus increasing the classifier's complexity. To reduce the number of features used by the classifier, we use a feature selection technique based on a genetic algorithm. Feature selection aims to select a subset of relevant and irredundant features that has high classification efficiency. We evaluate the selected feature subsets through the classification accuracy of a classifier trained using these feature subsets.

Feature extraction and feature sets: Feature extraction, as defined by Devijver and Kittler[10], is the problem of "extracting from the raw data the information which is most relevant for classification purposes, in the sense of minimizing the within-class pattern variability while enhancing the between-class pattern variability." Therefore, achieving a high recognition performance in an OCR system is highly influenced by the selection of efficient feature extraction methods, taking into consideration the domain of the application and the type of classifier used[11].

Any efficient feature extraction method should preferably possess two qualities: Invariance and reconstruct-ability[11]. Features that are invariant to certain transformations on the characters would be able to recognize many variations of these characters. Such transformations include translation, scaling, rotation, stretching, skewing and mirroring. On the other hand, the ability to reconstruct characters from their extracted features ensures that complete information about the character shape is present in these features. The importance of these two qualities for an efficient feature extraction method is accentuated for offline handwritten OCR systems. Such systems are the target of this research.

In this regard, two of the most widely-used feature sets in pattern recognition are the Normalized Central Moments (NCMs) and the Zernike moments.

Normalized central moments: The moments of order (u+v) of an image composed of binary pixels B(x, y) are found by[12,13]:

$$m_{uv} = \sum_{x}\sum_{y} x^{u} y^{v} B(x,y) \qquad u,v = 0,1,2,3,\ldots \tag{1}$$

As it can be shown from Eq. 1, m_{00} is the body's area A and the image's center of mass $(\bar{x},\bar{y})$ is found from:

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}} \tag{2}$$

The central moments, which are translation invariant, are found by:

$$\mu_{uv} = \sum_x \sum_y (x-\bar{x})^u (y-\bar{y})^v B(x,y) \qquad (3)$$

Finally, the normalized central moments, which are translation and scale invariant, are derived from the central moments as follows:

$$\eta_{uv} = \frac{\mu_{uv}}{(\mu_{00})^k} \qquad (4)$$

where, k = 1+(u+v)/2 for u+v≥2.

Zernike moments: Zernike polynomials are a set of complex polynomials which form a complete orthogonal set over the interior of the unit circle[14]. The form of these polynomials is:

$$V_{nm}(\rho,\theta) = V_{nm}(\rho,\theta) = R_{nm}(\rho)\exp(jm\theta) \qquad (5)$$

where $j=\sqrt{-1}, n \geq 0, n-|m|$ is even, $|m| \leq n$, ρ is the length of the vector from the origin to the point (x, y), θ is the angle between this vector and the x axis in the counterclockwise direction and the radial polynomial $R_{nm}(\rho)$ is:

$$R_{nm}(\rho) = \sum_{s=0}^{(n-|m|)/2} \frac{(-1)^s (\rho)^{n-2s}(n-s)!}{s!\left(\frac{n+|m|}{2}-s\right)!\left(\frac{n-|m|}{2})-s\right)!} \qquad (6)$$

Zernike moments are the projections of the image function onto these orthogonal basis functions. The Zernike moment of order n with repetition m for a digital image is given by:

$$A_{nm} = \frac{n+1}{\pi}\sum_x \sum_y B(x,y)[V_{nm}(\rho,\theta)]^* \qquad (7)$$

where, * is the complex conjugate operator and $x^2+y^2 \leq 1$.

To calculate the Zernike moments for a given image, its pixels are mapped to the unit circle, $x^2+y^2 \leq 1$. This is done by taking the geometrical center of the image as the origin and then scaling its bounding rectangle into the unit circle, as shown in Fig. 1.

Due to the orthogonality of the Zernike basis, the part of the original image inside the unit circle can be approximated using its Zernike moments A_{nm} up to a given order n_{max} using:

$$\hat{B}(x,y) = \sum_{n=0}^{n_{max}} \sum_m A_{nm} V_{nm}(\rho,\theta) \qquad (8)$$

where, $n-|m|$ is even and $|m| \leq n$.

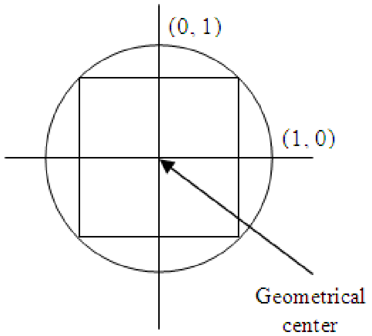

Fig. 1: Mapping an image rectangle into the unit circle

The orthogonality property of Zernike moments, as expressed in the previous equation, allows easy image reconstruction from its Zernike moments by simply adding the information content of each individual order moment.

Moreover, Zernike moments have simple rotational transformation properties[14]. Interestingly enough, the Zernike moments of a rotated image have identical magnitudes to those of the original one, where they merely acquire a phase shift upon rotation. Therefore, the magnitudes of the Zernike moments are rotation invariant features of the underlying image. Translation and scale-invariance, on the other hand, are obtained by shifting and scaling the image into the unit circle.

MATERIALS AND METHODS

Our experimental setup comprises a database of handwritten Arabic samples, feature extraction, feature selection and analysis tools.

Database of handwritten Arabic samples: Our database of handwritten Arabic samples was collected from 48 persons[15,16]. These persons were selected to represent various age, gender and educational background groups. The samples were collected by asking the participants to write, as they normally do, on a blank paper a one page of cursive Arabic text. This text was carefully selected so that it contains all the letter forms of the 28 basic Arabic letters.

We extracted from the 48 page samples collections of letter forms. Each collection comprises 48 samples from 48 different persons. Figure 2 shows the collection of 48 samples of the isolated Ain form.

The collections for initial, medial and final letter forms were extracted after manually segmenting their cursive sub words into individual letters. Manual segmentation is used to avoid errors that may come from an automatic letter segmentation process.

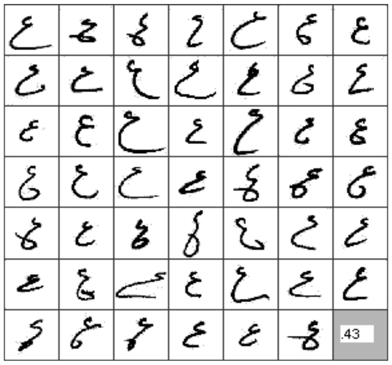

Fig. 2: A collection of 48 samples of the isolated Ain form

Automatic segmentation often suffers from over-segmentation, under-segmentation, or imprecise segmentation points positioning[17-19]. We use in this research 104 collections of letter forms: 30 isolated forms, 22 initial forms, 22 medial forms and 30 final forms. These collections contain all the basic 28 basic Arabic letters.

Secondary components detection: Detecting the secondary components can be done after segmenting the binary image of the letter into its disconnected components using the connected component labeling techniques[20]. Then the main body is easily identified as it is usually the largest component and is closer to the letter's center than the secondary components. After detecting the secondary components, the letter image is partitioned to main body and secondary components.

Feature extraction tool: To allow easy extraction of many features from the database of handwritten Arabic samples, we developed a feature extraction tool using C++ programming language under Microsoft Visual Studio development environment. In addition to preprocessing routines which include binarization, noise removal, thinning and boundary finding, various feature extraction routines were implemented in this application, including the normalized central moments and Zernike moments. These routines were applied on the 104 collections of letter forms and the results were exported for further analysis.

Using this tool, we extracted 52 Normalized Central Moments (NCM) of orders up to nine and 49 Zernike moments of orders up to 12. Three sets of these moments are extracted: from the whole letter's image, from the main body and from the secondary components.

Feature selection using NSGA: We have evaluated several feature selection techniques and decided to use the Non-dominated Sorting Genetic Algorithm (NSGA) for its superior results. NSGA is an efficient algorithm for multi-objective evolutionary optimization[21,22]. This algorithm searches for a set of optimal solutions of feature subsets among an evolving population of feature subsets. Best feature subsets evolve from one generation to the next.

We use a fast implementation of the multi-objective genetic algorithms (NSGAII) developed by Illinois Genetic Algorithms Laboratory[23]. The used parameter settings are as follows:

- Population size: 128
- Number of generations: 1000
- Selection type: Tournament of size 2 without replacement
- Crossover probability (p_c): 0.8 using simulated binary crossover and 0.8 gene-wise swap probability
- Probability of mutation (p_m): 0.1, selective

We used NSGA to search for optimal set of solutions with two objectives: (i) minimize the number of features m used in classification and (ii) minimize the classification error A. To evaluate the fitness of an individual, the NSGA program calls the SVM classifier described below. Given a subset of m features, the classifier returns the classification accuracy A.

To reduce execution time, we only used half of the available samples in the NSGA experiments. The 2-fold cross validation method was used to avoid over-fitting and to get stable results[24,25]. In a general k-fold cross validation method, the samples are split into k disjoint sets and training is repeated k times, each time with a different set held out as a validation set. The average accuracy of the k iterations is the reported accuracy A.

SVM classifier: We used the popular Support Vector Machine (SVM) classifier to evaluate the efficiency of the normalized central moments and the Zernike moments in the recognition of handwritten Arabic letters. SVM uses kernels to construct linear classification boundaries in higher dimensional spaces[26]. SVM selects a small number of critical boundary samples from each class and builds a linear discriminant function.

The SVM package used was the LIBSVM package[27]. Using grid search, we found that best results are achieved with the RBF kernel (radial basis function), penalty parameter C = 12 and gamma parameter y = 0.04. The data was first scaled to zero mean and one standard deviation.

RESULTS

For each sample in our database, the moments were calculated for the whole letter, its main body and its secondary components for a total of 52×3 = 156 NCMs and 49×3 = 147 Zernike moments. The magnitudes of the Zernike moments were used because of their rotation-invariance property. The extracted features were then fed to the NSGA for feature selection and finally the best feature subsets were evaluated using the SVM classifier using all samples and 5-fold cross validation.

Several experiments were carried out using this procedure. In the first experiment, the effect of adding the NCMs of the letter's main body and its secondary components to those of the whole letter was studied. The result is shown in Fig. 3 which illustrates the classification error as a function of the number of moments used. As it is obvious from Fig. 3, the classification error of the feature subsets selected from the mixture of moments is substantially smaller than that of the subsets selected from the whole-letter moments alone. For example, for a subset of 20 moments, the classification error is around 60% for the whole-letter moments while it is only 45% for the moments mixture.

A similar experiment was also done using the Zernike moments and the same result was obtained. As shown in Fig. 4, for a subset of 20 moments, the classification error of the moments mixture is approximately 10% less than that of the whole-letter moments.

To corroborate the previous results, a third experiment was performed in which the corresponding sets of moments from the previous two experiments were combined into an assortment of 52+49 = 101 NCMs and Zernike moments extracted from the whole letter and a second assortment of 101×3 = 303 moments extracted from the whole letter, its main body and its secondary components. The result is consistent with the preceding inferences. Figure 5 shows that selecting feature subset of size 20 from the moments of the whole letter and the constituents of the letter yields around 9% reduction in the classification error compared to selecting a feature subset from the moments of the whole body alone. Moreover, the recognition performance of the two types of moments together is better than the performance of each type individually (Fig. 3 and 4).

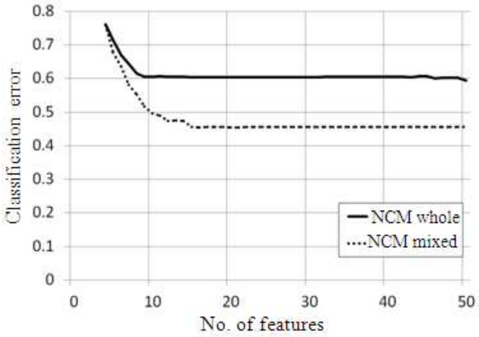

Fig. 3: NCMs classification error

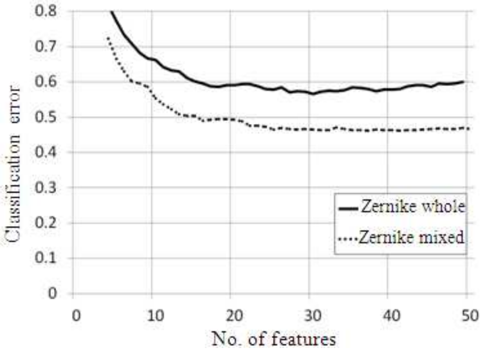

Fig. 4: Zernike moments classification error

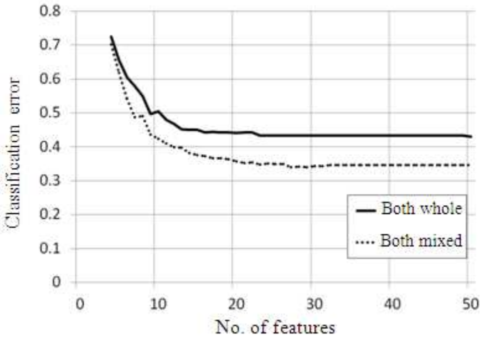

Fig. 5: Combined NCMs and Zernike moments classification error

DISCUSSION

Using moment features alone, as illustrated in the previous three figures, does not give classification error below 34%. However, moment features can be combined with other efficient feature extraction techniques to get high recognition accuracy. Figure 6 shows that a classification error of about 10% can be achieved when feature subsets are selected from the moment features and other efficient features that were studied in[28].

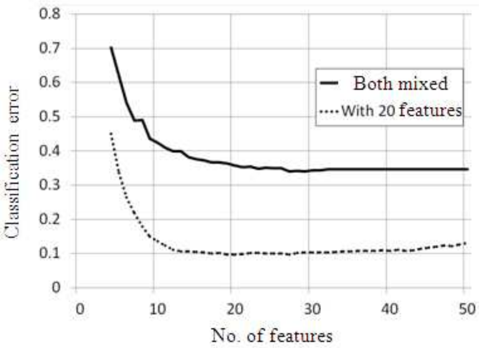

Fig. 6: Classification error of combined moments and 20 efficient features

These efficient features include the letter form, secondary type and position, some low-order elliptic Fourier descriptors and some statistical features extracted from the main body or the boundary, such as the area, orientation and perimeter to diagonal ratio.

The results shown in the previous four figures illustrate that higher recognition accuracies are achieved using the proposed feature extraction technique. Extracting features from the whole letter image, as well as, its main body and secondary components provides more valuable features that exploit the recognition potential of the secondary components of handwritten Arabic letters. These results also confirm the importance of the secondary components of the handwritten Arabic letters[8,9].

CONCLUSION

This study presented an approach for extracting features to achieve high recognition accuracy of handwritten Arabic letters. This approach exploits the classification potential of the secondary components of Arabic letters and overcomes some of their handwritten variations. This approach extracts moment features not only from the whole letter, but also from the main body and the secondary components.

The results presented in this study show that better recognition accuracies are achieved when features are selected from the mixture of moment features. This approach can be combined with other feature extraction techniques to achieve high recognition accuracy. We recommend using this approach when extracting moment features as well as other statistical features, such gradients and chain code descriptors.

ACKNOWLEDGMENT

This research was supported in part by the Deanship of Academic Research, The University of Jordan.

REFERENCES

1. Gordon, R.G., 2005. Ethnologue: Languages of the World. 15th Edn., SIL International, Dallas, ISBN: 10: 1-55671-159-X.
2. Arica, N. and F. Yarman-Vural, 2002. Optical character recognition for cursive handwriting. IEEE Trans. Patt. Anal. Mach. Intell., 24: 801-813. DOI: 10.1109/TPAMI.2002.1008386
3. Lorigo, L. and V. Govindaraju, 2006. Offline Arabic handwriting recognition: A survey. IEEE Trans. Patt. Anal. Mach. Intell., 28: 712-724. DOI: 10.1109/TPAMI.2006.102
4. Pechwitz, M., S.S. Maddouri, V. Märgner, N. Ellouze and H. Amiri, 2002. IFN/ENIT-database of handwritten Arabic words. Proceeding of 7th International Francophone Conference on Document Processing (CIFED 2002), Oct. 21-23, Hammamet, Tunis, pp: 129-136. http://www.ifnenit.com/download/CIFED_02_ifn-enit-database.pdf
5. Märgner, V., M. Pechwitz and H. ElAbed, 2005. Arabic handwriting recognition competition. Proceeding of 8th International Conference Document Analysis and Recognition, Aug. 29-Sept. 01, IEEE Xplore Press, USA., pp: 70-74. DOI: 10.1109/ICDAR.2005.52
6. Märgner, V. and H. El-Abed, 2007. Arabic handwriting recognition competition. Proceeding of International Conference Document Analysis and Recognition, Sept. 23-26, IEEE Xplore Press, Parana, pp: 1274-1278. DOI: 10.1109/ICDAR.2007.4377120
7. LIU, C.L., K. Nakashima, H. Sako and H. Fujisawa, 2003. Handwritten digit recognition: Benchmarking of state-of-the-art techniques. Patt. Recog., 36: 2271-2285. DOI: 10.1016/S0031-3203(03)00085-2
8. Abandah, G. and M. Khedher, 2009. Analysis of handwritten Arabic letters using selected feature extraction techniques. Int. J. Comput. Proc. Language., 22(1). (In press).
9. Khedher, M. and G. Al-Talib, 2005. Recognition of secondary characters in handwritten Arabic using fuzzy logic. Proceeding of the International Conference on Machine Intelligence, Nov. 5-7, ACIDCA, Tozeur, Tunisia, http://www.regim.org/conferences/acidca-icmi2005/proceedings.htm
10. Devijver, P.A. and J. Kittler, 1982. Pattern Recognition: A Statistical Approach. Prentice-Hall, London, ISBN: 10: 0136542360.

11. Trier, O.D., A.K. Jain and T. Taxt, 1996. Feature extraction methods for character recognition-a survey. Patt. Recog., 29: 641-662. http://cat.inist.fr/?aModele=afficheN&cpsidt=3045745
12. Theodoridis, S. and K. Koutroumbas, 2006. Pattern Recognition. 3rd Edn., Academic Press, ISBN: 10: 0123695317, pp: 856.
13. Reiss, T.H., 1991. The revised fundamental theorem of moment invariants. IEEE Trans. Patt. Anal. Mach. Intell., 13: 830-834. DOI: 10.1109/34.85675
14. Khotanzad, A. and Y.H. Hong, 1990. Invariant image recognition by Zernike moments. IEEE Trans. Patt. Anal. Mach. Intell., 12: 489-497. DOI: 10.1109/34.55109
15. Khedher, M.Z. and G. Abandah, 2002. Arabic character recognition using approximate stroke sequence. Proceeding of 3rd International Conference on Language Resources and Evaluation (LREC 2002), Arabic Language Resources and Evaluation-Status and Prospects Workshop, June 1-1, European Language Resources Association, Las Palmas, Canary Islands, Spain, pp: 28-34. http://www.abandah.com/gheith/kanary5.pdf
16. Abandah, G. and M. Khedher, 2004. Printed and handwritten Arabic optical character recognition-initial study. A report on research supported by the Higher Council of Science and Technology, Jordan. http://www.abandah.com/gheith/OCR_Report.pdf
17. Sari, T., L. Souici and M. Sellami, 2002. Off-line handwritten Arabic character segmentation algorithm: ACSA. Proceeding of International Workshop Frontiers in Handwriting Recognition, IEEE Xplore Press, USA., pp: 452-457. DOI: 10.1109/IWFHR.2002.1030952
18. Lorigo, L. and V. Govindaraju, 2005. Segmentation and pre-recognition of Arabic handwriting. Proceeding of 8th International Conference Document Analysis and Recognition, Aug. 29-Sept. 01, IEEE Xplore Press, USA., pp: 605-609. DOI: 10.1109/ICDAR.2005.207
19. Bentrcia, R. and A. Elnagar, 2008. Handwriting segmentation of Arabic text. Proceeding of 5th IASTED International Conference on Signal Processing, Pattern Recognition and Applications, Innsbruck, Feb. 13-16, ACTA Press, Austria, Canada, pp: 122-127. http://www.actapress.com/Abstract.aspx?paperId=32550
20. Rosenfeld, A. and A. Kak, 1976. Digital Picture Processing. 1st Edn., Academic Press, New York, ISBN: 10: 0125973608.
21. Srinivas, N. and K. Deb, 1995. Multi-objective function optimization using non-dominated sorting genetic algorithms. Evolut. Comput., 2: 221-248. DOI: 10.1162/evco.1994.2.3.221
22. Zitzler, E., K. Deb and L. Thiele, 2000. Comparison of multiobjective evolutionary algorithms: Empirical results. Evolut. Comput., 8: 173-195. DOI: 10.1162/106365600568202
23. Deb, K., A. Pratap, S. Agrawal and T. Meyarivan, 2002. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. IEEE Trans. Evolut. Comput., 6: 182-197. DOI: 10.1109/4235.996017
24. Kohavi, R., 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceeding of 14th International Joint Conference on Artificial Intelligence, (IJCAI’95), Standfard, pp: 1137-1143. http://robotics.stanford.edu/~ronnyk/accEst.ps
25. Stone, M., 1974. Cross-validatory choice and assessment of statistical predictions. J. R. Stat. Soc., 36: 111-147. http://www.jstor.org/pss/2984809
26. Burges, C., 1998. A tutorial on support vector machines for pattern recognition. Knowl. Discov. Data Mining, 2: 1-43. DOI: 10.1023/A:1009715923555
27. Hsu, C.W. and C.J. Lin, 2002. A comparison of methods for multi-class support vector machines. IEEE Trans. Neural Network., 13: 415-425. DOI: 10.1109/72.991427
28. Abandah, G., K. Younis and M. Khedher, 2008, Handwritten Arabic character recognition using multiple classifiers based on letter form. Proceeding of 5th IASTED International Conference on Signal Processing, Pattern Recognition and Applications, Innsbruck, Austria, Feb. 13-16, ACTA Press, Canada, pp: 128-133. http://www.actapress.com/Abstract.aspx?paperId=32551