

Multiple Classifiers and Invariant Features Extraction for Digit Recognition

Derdour Khedidja*, Mouss Hayet

Industrial Engineering Departement, Automatic and Manufacturing Laboratory, University of Batna 2, Algeria.

* Corresponding author. Tel.: (213) (0) 05 52 20 79 90; email: der_kh77@yahoo.fr

Manuscript submitted June 30, 2018; accepted September 20, 2018.

doi: 10.17706/ijcee.2019.11.1.41-52

Abstract: Extraction of feature and classification methods are important phases in recognition system. A good classifier and extraction of features that suits play a very important role in a recognition system to improving recognition rate. In this paper we propose a new system designed to recognize the ten digits of printed Arabic numerals that are the most common symbolic representation of numbers in the world today. This study has been conducted using Hu moment, number of hole and surface which are tolerate to the geometric transformations along with seven different classifiers Naive Bayesian, Multi-Layer Perceptron (MLP), Linear Discriminant Analysis (LDA), Pseudo-Inverse. Support Vector Machine (SVM), Decision Tree and K-Nearest Neighbor (KNN). Classifier combination is considered. Experimental tests demonstrated that our technique achieves good results on multi-font and multi-size printed digit dataset.

Key words: Classification, features extraction, printed digit recognition, cross-validation, multiple classifiers, Hu moments.

1. Introduction

The printed digit recognition is an important area of research in OCR application pattern classification and other organization, for its applications in car plates like ANPR (Automatic number plate recognition), post office (postal address, postal sorting), barcode, pin code [1] and street view imagery [2]. Much research has been done on printed digit recognition in different language (Arabic, Indian, Persian/Farsi, Roman, Chinese, Odia, Bangla, Devanagari, Hindu, Telugu, Urdu etc.) [3]-[12], though there are some commercially available software. In recent years, various applications replace the handwritten documents by printed content. In the academics and production fields the recognition of printed digits has received an extensive attention, and a big progress has been known in numerous industrial applications.

The present research work in this paper is to develop a recognition system of printed digit. One of the most widely studied topics in the field of pattern recognition and machine learning is combination of multiple classifiers, this mechanism preserves the individual strength of classifiers and compensate their individual weakness, it has shown their effectiveness to combine results received by different classifiers. The proposed system recognizes the printed digit by multiple classifiers system using KNN, SVM, decision tree, pseudo inverse, LDA, Naive Bayesian and MLP. Hu moments features which it is tolerate to the geometric transformations (translation, scaling and rotation) were extracted and improved by hole number with surface and east cavity of digit. The experimental tests that have been performed on the multi-font and multi-size printed digits, in which not much work has reported in recent times on the recognition of

multi-font and multi-size digits. The results demonstrate the effectiveness of the proposed solution. And here, we have made an attempt to combine the decision of classifiers to achieve the recognition.

The rest of the paper is organized as follows: Section 2 explores past work on digit recognition and on combination of classifiers. Sections 3 describe the proposed method and Section 4 present the feature set used. In Section 5 we describe the several classifiers implemented. Experimental results show in Section 6. Conclusion has been drawn in Section 7.

2. Related Works

In literature, many different approaches and methods have been widely used to recognize printed and handwritten numerals. A survey of recent publication in numeral recognition for Urdu, Arabic, and Farsi is presented in [13]. Authors in [14] employed the correlation factor in the process of recognition of offline machine-print decimal number; they calculated the correlation coefficient between certain reference images and the image under consideration. In [15] authors used ten different classifiers like Linear classifier, Artificial Immune system, Euclidean distance, Chebyshev distance, Cosine distance, Manhattan distance, KNN, K-means, K-medoids, with 8 features different computed from image fusion, zoning extraction, fan beam projections, radon transform, discrete Fourier transform, curvelet transform, directional chain code and run length count then classifier fusion are considered, for recognition of handwritten Kannada numerals.

Reference [1] proposes a new technique of hybrid classification for recognition printed numerals. Fourier Descriptors (FD), object region boundary analysis, and Chain code based algorithm used to perform The feature extraction. Authors proposed A new curve tracing Chain code based algorithm (CTCC) were the curve features has been extracted from the digit images. Dynamic programming and MLP with back propagation algorithm (MLP-BP) used to perform the recognition. Higher accuracy of 99% was obtained.

A new method is proposed by [16], of handwritten pre-processing and printed and isolated numeral characters. Were the size of the input image of each numeral was reduced by discarding the redundant information. this method reduces also the number of features of the attribute vector provided by the extraction features method. Recognition of digit is done KNN and MLP classifiers. For Arabic handwritten and printed digit recognition, another new system using the characteristics loci (CL) is presented by [17] for extracting features of digit, the recognition is carried out through KNN and MLP techniques.

Much research has been done on printed and handwritten digit recognition using artificial neural network: [10], [12], [13], [18]-[28]. Recently, researchers have begun to pay attention to improve the performance by integration of multiple classifier such as [1], [9], [29]-[37].

3. Recognition System Architecture

The architecture of our proposed system shown in "Fig. 2", where multiple classifiers treat each digit. For preparation of the training and the test sets of samples, a database of 6240 digit samples is formed by digits (0-9) in ten images, like the image of digit "3" shown in "Fig. 1". Each one of image contains 624 digit instances in different sizes and fonts. we have used different size and The majority office word fonts The digit images It was stored in Portable Network Graphics (PNG) format by paint application of windows. The binarisation of image was performed using threshold method and each image binarised (normalised) were segmented into separate individual digit and locate each digit by drawing a rectangle around the digit (tracing boundary). Each image digit is used as input for the schema used to test the system proposed.

In the field of pattern recognition there exist a significant problem, it is the recognition of objects regardless of their position, size and orientation. For this, using of the invariant moments has been proposed, we perform printed digit recognition experiments using invariant feature set calculated from Hu

moments and holes number.

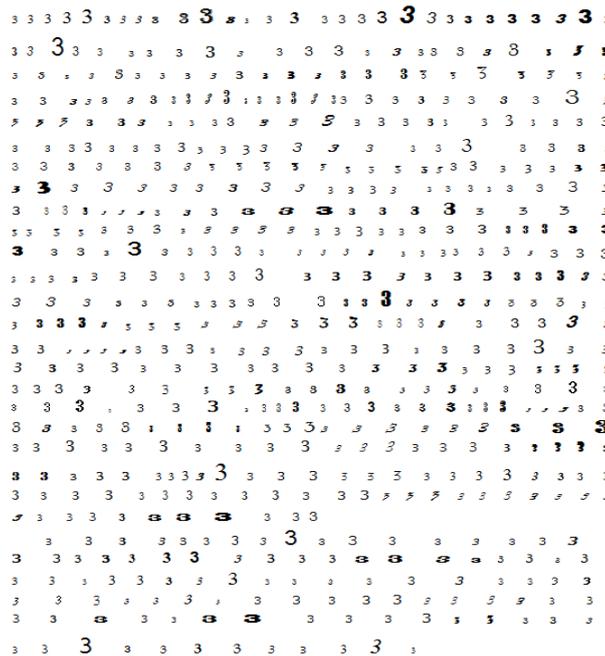


Fig. 1. Image of digit “3”.

Integration of classifiers goal by [38] is to generate more certain, precise and accurate system results, many methods have been proposed in supervised learning to construct a good ensembles of classifiers. mechanisms that are used to build ensemble of classifiers include: 1) using different subsets of training data with a single learning method. 2) using different training parameters with a single training method. 3) using different learning methods (classifiers).

In this paper, seven different sets of classifiers are used KNN classifier based on Square Euclidean distance, SVM, decision tree, pseudo-inverse, LDA, naive Bayesian and MLP. We use "over produce and choose paradigm" it consist to combine all classifiers with Major Voting combination method which is a simple way to combine multiple classifiers, and try to determine which subset of classifiers selected among the above 7, achieve the optimal combination results.

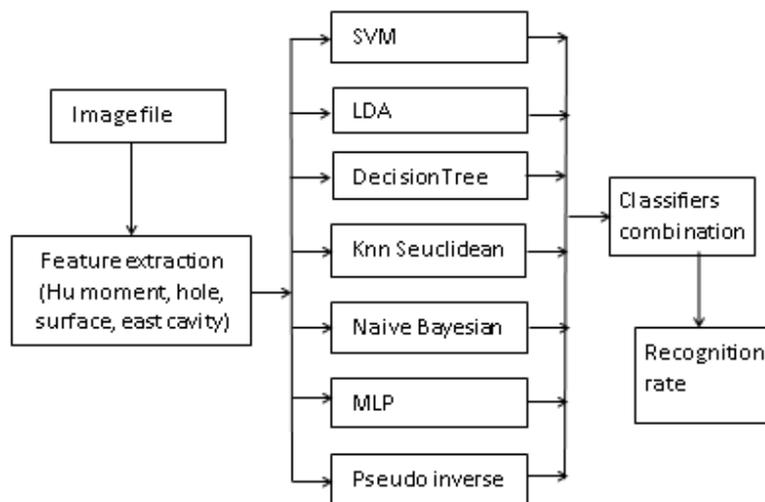


Fig. 2. The framework of proposed digit recognition method.

4. Feature Extraction

Feature extraction is an important, crucial and delicate process for a good recognition of digits. It consists to transform the image into set of characteristics presented in attribute vector, which make the recognition an easy task. authors in [39] can be broadly classified Features into two different categories: structural features (like strokes, intersections of line segments, end points, loops and stroke relations) and statistical features (like, moments, n-tuples, zoning, characteristic loci ...). In the present work, we employed Hu moment features.

There are seven Hu moments proposed by [40] and mentioned in [41], there are well known to be invariant to size, orientation and position, of digit. They are pure statistical measures of the pixel distribution around the center of gravity of the character. They are the following:

$$\phi_1 = \eta_{20} + \eta_{02} \tag{1}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{2}$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{3}$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} - \eta_{03})^2 \tag{4}$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03}) + (\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \tag{5}$$

$$\phi_6 = (\eta_{20} + \eta_{02})((\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2) + 4\eta_{11}(\eta_{03} + \eta_{12})(\eta_{21} + \eta_{03}) \tag{6}$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} + 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \tag{7}$$

where η_{ij} are the normalized Geometrical central moments, which are invariant to scaling (scale normalized).

$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{(i+j+2)/2}} \tag{8}$$

$$\mu_{ij} = \sum_{k=1}^K \sum_{l=1}^L (k - x_g)^i (l - y_g)^j g(k, l) \tag{9}$$

where $g(k,l)$ is image function and K, M are image dimensions, and μ_{ij} are Geometrical central moments of order $(i+j)$ computed using centroid (gravity center) which calculated by (10). μ_{ij} are invariants to translation (with image translation to coordinate origin while computing central moments).

$$x_g = \frac{M_{10}}{M_{00}}, y_g = \frac{M_{01}}{M_{00}} \tag{10}$$

where M_{ij} are geometrical moments of order $(I+j)$ for binary image defined as :

$$M_{ij} = \sum_{k=1}^K \sum_{l=1}^L (k)^i (l)^j g(k, l) \tag{11}$$

“Table 1” lists moment invariants for different status of digits “5”

	ϕ_1	ϕ_2	ϕ_3	ϕ_4	ϕ_5	ϕ_6	ϕ_7
	0.3595	0.0210	0.0012	0.0002	0.0000	0.0000	-0.0000
	0.3595	0.0210	0.0012	0.0002	0.0000	0.0000	-0.0000

5 5 5 5 5 5	0.3595	0.0210	0.0012	0.0002	0.0000	0.0000	-0.0000
	0.3595	0.0210	0.0012	0.0002	0.0000	0.0000	0.0000
	0.3595	0.0210	0.0012	0.0002	0.0000	0.0000	0.0000
	0.3258	0.0039	0.0002	0.0000	-0.0000	0.0000	0.0000
	0.3345	0.0111	0.0003	0.0000	0.0000	0.0000	0.0000
	0.3345	0.0111	0.0003	0.0000	-0.0000	-0.0000	0.0000

5. Classification

The classification is done using 7 classifiers: SVM, LDA, KNN classifier based on Square Euclidean distance, decision tree, pseudo-inverse, Naive Bayesian and MLP, presented in the follow:

5.1. SVM Classifier

SVM classifier can transform a problem nonlinear separable into a linear separable problem by projecting data into the feature space and then finding the optimal separate hyper plane. Initially this method was proposed to solve two class problems. Later, a few approaches were proposed in [42] to extend this technique to multiclass classification problems. Two strategies are currently used to solve the problems of multiclass, Building binary classifiers that distinguish between every pair of classes (one-versus-one) or between one of the class and the rest (one-versus-all).

5.2. KNN Classifier

It is one of the powerful and simplest method for classification, which through a training set of prototype allows an almost infallible classification of a new prototype. The method thus uses two parameters, the number K and the distance function to compare the new case to cases already classified more details is given by [38].

5.3. Naïve Bayesian

It based on the application of Bayes theorem with strong (naïve) independence hypotheses between the features. Naive Bayes classifier supposes that the particular feature value is unrelated to the absence or presence of any other feature, given the class variable. the major advantage of the naive Bayes classifier by [38] is its short computational time for training.

5.4. Pseudo inverse

This method defined in [43] needs firstly a learning phase. During this phase, it computes a linear transformation applied to the input, the result provided a vector close to the desired output vector, it allows to guess the class.

5.5. LDA

LDA by [44] it used in machine learning and pattern recognition to find a linear combination of features, which separates or characterizes two or more classes of objects. The principle consists to approximating boundaries between classes in the m-dimensional feature space by placing hyper planes, among regions corresponding to the different classes, assuming them to be linearly separable.

5.6. Decision Tree

the classification is achieved with decision tree basing on series of tests on the features. It is a simple idea, through the order of these tests the response at one of them indicates what next test, which this object must

be submit. The principle is the organization of all possible tests as a tree [2]. A classes are denoted the leafs of this tree and the test associate a node. And [38] described decision trees are trees that classify instances by sorting them based on feature values. each node in a decision tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume. instances are classified starting at the root node and sorted based on their feature values.

5.7. MLP Classifier

The used type of neural network is a feed-forward artificial neural network model; it is the most widely known. It consists of one input layer, multiple (one or more in our case one layer) hidden layers of neurons and one output layer, with each layer fully connected to the next one. In the input layer the Number of neurons is equal to the length of the features vector, but in the hidden layers is decided experimentally and in the output layer the number of neurons is equal to the number of classes, more description presented in [38].

6. Experimental Results

For the evaluation of our system we used for learning three techniques of cross validation, it consists to partition the sample of data into complementary subsets. the set called the training set is the subset were we Perform the analysis, and we validate the analysis on the other subset called the validation set or testing set. In our experiments, we used "test set validation" (called holdout method), "k-fold cross-validation", and "Resubstitution method".

In the method called k-fold cross-validation (k-foldcv), k=10 is commonly used, the original sample partitioned randomly into k equal size subsamples. One subset retains for testing the model as the validation data, and the training data is the remaining (k-1) subsamples. The process of cross-validation is then repeated k times. The advantage of this method is that all observations are used for both training and validation, and each observation is used for validation exactly once, the second method divides the database of size n in training sample and test sample. The third, Resubstitution method uses all samples as training and testing data. for each technique of cross validation, we use the stratification, this means that each fold contains roughly the same proportions of the all types of class labels.

Table 2. Performance of 7 Classifiers by 7 Hu Moments

Classifiers	10-fold CV		Holdout		Resubstitution
	Non stratified	stratified	Non stratified	stratified	
Pseudo-inverse	33.13	33.00	32.59	33.01	32.98
LDA	40.53	37.23	40.49	39.00	40.88
N_Bayes	46.81	46.60	44.44	48.08	55.80
Decision tree	65.99	60.71	63.78	61.65	86.35
SVM	32.36	31.49	37.82	36.00	29.57
KNN	74.52	73.24	69.87	75.00	99.57
MLP	58.11	57.53	55.24	56.84	57.88
Major voting of all classifiers	62.98	58.97	58.12	64.64	74.63

Initially, each classifier is trained independently and separately using Hu moment features, Experimental results show in Table 2 for improving results of each classifier we combine Hu moment with hole numbers feature were Table 3 present the results. The 5 best classifiers are: KNN, decision tree, MLP, N-Bayes and

LDA, by order for the three learning technique, the best recognition rate is obtained by KNN classifier for three techniques.76.76%, 73.61%, 99.57% respectively. We note that the stratification has no great effect to improve classifier performance.

Table 3. Performance of 7 Classifiers by Hu Moment and Hole Number

Classifiers	10-fold CV		Holdout		Resubstitution
	Non stratified	stratified	Non stratified	stratified	
Pseudo-inverse	35.34	35.50	39.10	34.94	35.50
LDA	45.46	41.94	47.76	44.55	45.93
N_Bayes	47.77	48.01	48.18	48.61	56.78
Decision tree	69.28	64.01	67.84	64.10	88.57
SVM	41.06	44.95	35.47	43.69	46.35
KNN	76.76	76.76	73.61	72.54	99.57
MLP	64.52	64.02	65.81	61.97	64.71
Major voting of all classifiers	72.43	71.31	69.23	69.66	80.29

We see the remarkable difference rate recognition between resubstitution technique with Kfold and holdout technique. The reason for this might be due by the numerals "6" and "9" are reverse of each other; were Hu moments cannot discriminate these digits with tolerance to the rotation transformation. To solve this problem, we proposed to add new features to distinguish between "6" and "9" by extracting east cavity shown in "Fig. 3", and to distinguish between "2","5" we calculate the normalized surface and the ratio height / width of each digits and, the recognition rate of the individual classifier shown in Table 4, the best results obtained by KNN classifier are 79.97%, 77.35%, 99.89% respectively by the three technique cross validation. Furthermore, the recognition rate of SVM classifier is increased from 46.35% for resubstitution method to 67.11%.

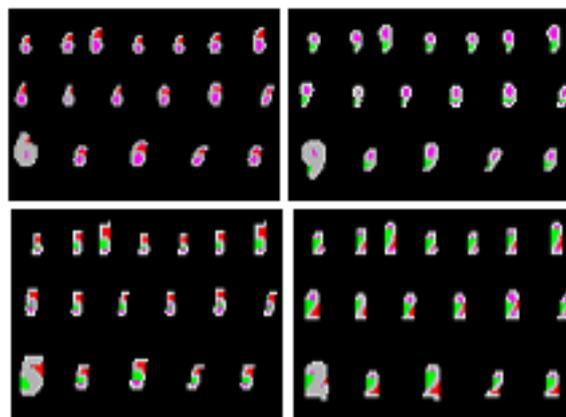


Fig. 3. The Est cavity in red color and hole in rose color features.

In order to see the combined effect, we use major voting for combining all classifier. Table 4 give the results 78.67%, 78.85%, 85.14% respectively for each technique, we note that the over production (combination of all classifiers) did not improve the performance of recognition system. The rate recognition is almost the same for k-fold and holdout technique, and decrease from 99.89% to 85.14% by resubstitution technique.

Table 4. Performance of 7 Classifiers by Hu Moment hole Number East Cavity. Surface and Ratio

Classifiers	10-fold CV		Holdout		Resubstitution
	Non stratified	stratified	Non stratified	stratified	
Pseudo-inverse	55.50	52.95	55.24	54.38	53.08
LDA	52.84	50.99	54.81	49.25	54.38
N_Bayes	54.96	54.66	54.91	54.70	63.78
Decision tree	75.93	69.46	74.15	66.88	91.14
SVM	62.08	62.02	60.04	63.78	67.11
KNN	79.97	79.65	73.93	77.35	99.89
MLP	73.03	71.96	70.51	72.12	72.84
Major voting of all classifiers	78.67	75.80	74.68	78.85	85.14

Finally, we try to choose the optimal subset of classifiers. The overall classification results are listed in Table 5. As can be seen that the better results are obtained with rate recognition of 82.69%, 80.88% and 94.44% on Knn-MLP-decision tree subset; and 78.84% 79.27, 95.22 on KNN -Decision tree -N-Bayes subset classifiers. we noticed that KNN-MLP-decision tree subset classifier is the better when compared the best rate recognition obtained with KNN classifier by cross-validation and holdout techniques.

Table 5. Performance of 7 Classifiers Combination by Hu Moment and Hole Number and East Cavity. Surface and Ratio

Classifiers	10-fold CV		Holdout		Resubstitution
	Non stratified	stratified	Non stratified	stratified	
KNN+SVM+ tree+bayes	81.25	76.12	73.93	82.37	92.00
KNN+SVM+tree+MLP+bays	81.25	77.24	75.96	82.05	89.63
KNN+SVM+tree+MLP	81.57	75.96	74.04	83.01	86.89
Knn+ MLP+tree+Bayes	80.77	80.29	78.74	80.77	93.11
KNN+SVM+tree	79.97	75.80	74.77	81.30	94.47
KNN+MLP+tree	81.57	82.69	78.74	80.88	94.44
KNN+SVM+N-bayes	73.08	68.75	67.73	72.11	81.73
KNN+SVM+MLP	76.92	71.95	70.83	78.09	82.13
KNN+decision tree+N-bayes	77.88	78.84	74.97	79.27	95.22
SVM+tree+bayes	70.99	69.71	66.66	71.79	78.94
SVM+MLP+tree	77.72	74.36	72.97	76.92	80.16
MLP+tree+Bayes	76.44	76.60	74.14	76.49	83.38
KNN+SVM	59.61	60.86	61.54	55.88	68.62
KNN+bayes	68.59	66.03	66.99	69.65	84.89
KNN+tree	76.44	73.08	72.65	77.14	94.70
KNN+LDA	66.19	63.14	62.71	66.56	77.03
KNN+pseudo_inv	65.87	63.47	65.17	67.09	77.59

Classifiers	10-fold CV		Holdout		Resubstitution
	Non stratified	stratified	Non stratified	stratified	
KNN+MLP	75.80	72.11	72.86	75.53	84.88
SVM+tree	58.81	59.61	59.83	55.12	67.09
SVM+bayes	54.32	57.53	56.51	49.68	63.16
Tree+bayes	68.75	66.82	66.88	69.34	79.95
Tree+MLP	75.96	71.63	72.22	74.79	81.09
MLP+SVM	58.33	62.98	61.85	54.17	65.51

Our technique obtained significantly high recognition rates 99.89% compared with 99% the results obtained in [1] where the system was tested with very less training samples -700- samples containing 70 set of each digit (0-9) of varying size and font style. The samples contain various fonts like Arial, Times New Roman, Courier New etc. as for our system was tested with a dataset of 6240 digit samples containing 624 set of Each digit (0-9) with varying size and majority office word font style.

7. Conclusion

In this paper we have presented a model for printed digit recognition. This model consists of two systems:

1- Invariant features extraction.

2- Classification system taking into consideration the different learning techniques, the partitioning of data and the combination of classifiers

This model doesn't an innovation in the field of numeral recognition; it is a contribution: on one hand contribution, is the reduction of features number and tolerance to the scale, shift and rotation, which can efficiently recognize patterns regardless of their possible position, rotation and size variations. the proposed method achieved higher recognition rate 99.89% considering less number of features, this result shows very satisfactory recognition performances when compared to the method in [1] with results obtained is 99%, and to Odia digit recognition rate (96.08%) in [27]. Through this results we can say that printed Arabic digits are actually easier to recognize than printed digits of other language. On the other hand, the combination of different classifiers can improve the recognition rate. A powerful classifier such as SVM and MLP may yield different recognition rate using different or same patterns features, but The high recognition rate can be obtained by selecting the low complexity classifier (K-NN) and suitable features extraction.

The final result is produced not only from the combination of the good classifier but also by the features it even. The improvement in features helps to attain high recognition accuracy. This method can also be tried for handwritten digits, and for further performance, there are series of improvements that could be made to this work; A first direction of improvement would be suggestion of using a large amount of training data, using other type of features for example Zernike moments which are tolerates to the geometric transformations and exploiting other mechanisms of classifier combination.

References

- [1] Radha, R., & Aparna, R. R. (2014). Digit recognition using hybrid classifier. *Proceedings of IEEE the World Congress on Computing and Communication Technologies (wccct)* (pp. 34-38). Trichirappalli,

India.

- [2] Cornuéjols, A., & Miclet, L. (2002). *Apprentissage Artificiel*. Eyrolles.
- [3] Hassanpoura, H., Samadiani, N., & Akbarzadeh, F. (2017). A modified self-organizing map neural network to recognize multi-font printed, Persian numerals. *International Journal of Engineering*, 30(11), 1700-1706.
- [4] Helali, M., Alneghaimish, A., & Ahmad, I. (2017). Handwritten digit recognition under constrained training conditions. *Proceedings of the 8th International Conference of Pattern Recognition Systems*.
- [5] Malallah, F. L., Saeed, M. G., & Aziz, M. M. (2016). Off-line Arabic (Indian) numbers recognition using expert system. *International Journal of Advanced Computer Science and Applications*, 7(4), 397.
- [6] Naz, S., Ahmed, S. B., Ahmad, R., & Razzak, M. I. (2016). Arabic Script based digit recognition systems (survey). *Proceedings of the International Conference on Recent Advances in Computer Systems*.
- [7] Mohapatra, R. K., Majhi, B., & Jena, S. K. (2016). Printed Odia digit recognition using finite automaton. *Proceedings of the 3rd International Conference on Advanced Computing, Networking and Informatics. Smart Innovation, Systems and Technologies: Vol 43*. Springer, New Delhi.
- [8] Jeiad, H. A. (2018). Indian number handwriting features extraction and classification using multiclass SVM. *Engineering and Technology Journal*, 36(1).
- [9] Moradi, V., Razzazi, F., & Behrad, A. (2016). Recognition of handwritten Persian two-digit numerals using a novel hybrid SVM/HMM algorithm. *Majlesi Journal of Electrical Engineering*, 10(3).
- [10] ElHindi, K., Khayyat, M., & Abu-Kar, A. (2016). Comparing the machine ability to recognize hand-written Hindu and Arabic digits. *Intelligent Automation & Soft Computing*.
- [11] Alom, M. Z., Sidike, P., Taha, T. M., & Asari, V. K. (2017). Handwritten Bangla digit recognition using deep learning. *Computer Vision and Pattern Recognition*, Cornell University. arXiv preprint arXiv: 1705.02680.
- [12] Singh, P. K., Sarkar, R., & Nasipuri, M. (2016). A Study of moment based features on handwritten digit recognition. *Applied Computational Intelligence and Soft Computing*.
- [13] Taqi, A. K., & Korial, A. E. (2018). Comparison between feature based and deep learning recognition systems for handwriting Arabic numbers. *Computer Engineering and Intelligent Systems*, 9(4).
- [14] Alqudah, A. T., Hussein, Al-Zoubi, R., & Al-Khassaweneh, M. A. (2012). Shift and scale invariant recognition of printed numerals. *Basic Sci. & Eng.*, 21, 41-49.
- [15] Mamatha, H. R., & Srikantamurthy, K. A. (2013). Comparative study of different feature extraction and classification methods for recognition of handwritten Kannada numerals. *International Journal of Database Theory and Application*, 6(4).
- [16] Benchaou, S., Nasri, M., El-Melhaoui, O., & Bouali, B. (2014). New approach of preprocessing for numeral recognition. *Int. Journal of Engineering Research and Applications*, 4, 26-30.
- [17] El-Melhaoui, O., El-Hitmy, M., & Lekhal, F. (2011). Arabic numerals recognition based on an improved version of the loci characteristic. *International Journal of Computer Applications*, 24, 975-8887.
- [18] Al-Omari, K. S., Putra, S., Al-Taweel, A. S., & Anas-Husain, J. A. (2009). Digital recognition using neural network. *Journal of Computer Science*, 5(6), 427-434.
- [19] Amritpal, K., & Madhavi, A. (2013). Neural network based numerical digits reorganization using NNT in MATLAB. *International Journal of Computer Science & Engineering Survey*, 4(5).
- [20] Derdour, K., & Mouss, H. (2015). Printed digits recognition using multiple multi-layer perceptron classifiers and Hu moments. *Proceedings of the Symposium on Complex Systems and Intelligent Computing (CompSIC)*.
- [21] Ian, J., Bulatov, Y., Ibarz, J., Arnoud, S., & Shet, G. V. (2014). Multi-digit number recognition from street view imagery using deep convolutional neural networks. *Computer Vision and Pattern Recognition*.

- [22] Shilbayeh, N. F., Alwakeel, M. M., & MohyNaser, M. (2013). An efficient neural network for recognizing gestural Hindi digits. *American Journal of Applied Sciences*, 10(9), 938-951.
- [23] Singh, D., & Singh, K. B. (2011). Digit recognition System using back propagation neural network. *International Journal of Computer Science and Communication*, 2, 197-205.
- [24] Sumedha-Hallale, B., & Geeta-Salunke, D. (2013). Offline handwritten digit recognition using neural network. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2, 4373-4377.
- [25] Venkata, R. K., Rajeswara, R. D., Ankaiah, U., & Rajesh, K. (2013). Handwritten character and digit recognition using artificial neural networks. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3(4).
- [26] Vijaya, K. R. R., & Ravi, B. U. (2018). Efficient handwritten digit classification using user-defined classification algorithm. *International Journal on Advanced Science Engineering Information Technology*, 8(3).
- [27] Ashiquzzaman, A., & Tushar, A. K. (2017). Handwritten Arabic numeral recognition using deep learning neural networks. *Proceedings of the 2017 IEEE International Conference on Imaging, Vision & Pattern Recognition*.
- [28] Arbain, N. A., Azmi, M. S., Muda, A. K., Muda, N. A., & Radzid, A. R. (2018). Offline handwritten digit recognition using triangle geometry properties. *International Journal of Computer Information Systems and Industrial Management Applications*, 10, 87-97.
- [29] Alimoglu, F., & Alpaydin, E. (2001). Combining multiple representations for pen-based handwritten digit recognition. *Turk. J. Elec. Engin.*, 9.
- [30] Bao, Y., Ishii, N., & Du, X. (2004). Combining multiple k-nearest neighbor Classifiers using different distance functions. *Intelligent Data Engineering and Automated Learning – IDEAL Lecture Notes in Computer Science*, 3177, 634-641.
- [31] Bay, S. D. (1998). Combining nearest neighbor classifiers through multiple feature subsets. *Proceedings of the 17th Intl. Conf. on Machine Learning* (pp. 191-209).
- [32] Derdour, K., & Mouss, H. (2015). Digit recognition using multiple classifiers. *Proceedings of the 12th IEEE International Symposium Programming and Systems (ISPS)*, (pp. 1-6).
- [33] Gorgevik, D., Cakmakov, D., & Radevski, V. (2001). Handwritten digit recognition by combining support vector machines using rule-based reasoning. *Proceedings of the 23rd Int. Conf. on Information Technology Interfaces*. Pula, Croatia.
- [34] Kumawat, P., Khatri, A., & Nagaria, B. (2013). Comparative analysis of offline handwriting recognition using invariant moments with HMM and combined SVM-HMM classifier. *Proceedings of the IEEE International Conference on Communication Systems and Network Technologies*.
- [35] Kim, J., Seo, K., & Chung, K. A. (1997). Systematic approach to classifier selection on combining multiple classifiers for handwritten digit recognition, document analysis and recognition. *Proceedings of the Fourth International Conference on Document Analysis and Recognition: Vol 2*.
- [36] Kumar, S. (2011). ACO based feature subset selection for multiple k-nearest neighbor classifiers. *International Journal on Computer Science and Engineering*, 3.
- [37] Prasad, B. K., & Sanyal, G. (2016). Novel features and a cascaded classifier based Arabic numerals recognition system. *Multidim. Syst. Sign Process*.
- [38] Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. *Informatica*, 31(2007), 249-268. IOS press.
- [39] Govindan, V. K., & Shivaprasad, A. P. (1990). Character recognition a review. *Pattern Recognition*, 23, 671-683.

- [40] Hu, M. K. (1962). Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 179-187.
- [41] Boveiri, H. R. (2011). Transformation-invariant classification of Persian printed digits. *International Journal of Signal Processing, Image Processing, and Pattern Recognition*, 4, 153-164.
- [42] Niun, X.-X., & Suen, C. Y. (2012). A novel hybrid CNN-SVM classifier for recognizing handwritten digits. *Pattern Recognition*, 1318-1325.
- [43] Burel, G. (2001). *Introduction Au Traitement D'Images Simulation Sous Matlab*. Hermes Science Publications.
- [44] Sankar, K., & Amita, P. (2001). Pattern recognition from classical to modern approaches. *World Scientific*.

Khedidja DERDOUR was born in Batna, Algeria, in 1977. She received her engineer degree in computer science from Batna University in 2001 and her magister in Industrial Engineering from the same university in 2010. Currently she prepares her doctorate degree in industrial engineering.

Her research interests include machine learning, classification, pattern recognition and image processing. She is interested in improving the classification accuracy and developing new features. She presented papers in five international conferences.

Leila Hayet MOUSS was born in Batna, Algeria, in 1954. She received the B.Sc. degree in electrical engineering, in 1979, from the National Polytechnic School of Algeria, Algeria; the M.Sc. degree in electrical and computer engineering, in 1982, from the ENSERB, France; and finally the Ph.D. degree also in electrical and computer engineering, in 1985, Bordeaux University, France.

After graduation, she joined the University of Batna, Algeria, where he is a professor of electrical and computer engineering. She possesses an experience of more than 30 years in the field of teaching and research. She has guided nearly 20 theses. Her current research interests include industrial diagnosis of production system using the artificial intelligence technique in the LAP Lab (Laboratoire d'Automatique et Productique) at Batna.

Prof. Mouss is a member New York Science Academy. She is the head of Automatic and Computer Integrated Manufacturing Laboratory.