Handwritten digit recognitiopn using rotations

Anca Ignat
Faculty of Computer Science
University "Alexandru Ioan Cuza" of Iași
Iași, Romania
ancai@info.uaic.ro

Bogdan Aciobăniței
Faculty of Computer Science
University "Alexandru Ioan Cuza" of Iași
Iași, Romania
bogdan.aciobanitei@info.uaic.ro

***Abstract*—Handwritten digit recognition is a subproblem of the well-known optical recognition topic. In this work, we propose a new feature extraction method for offline handwritten digit recognition. The method combines basic image processing techniques such as rotations and edge filtering in order to extract digit characteristics. As classifiers, we use *k*-NN (*k* Nearest Neighbor) and Support Vector Machines (SVM). The methods are tested on a commonly employed database of handwritten digits' images, MNIST (Mixed National Institute of Standards and Technology) on which the classification rate is over 99%.**

Keywords—handwritten digit recognition, image rotations, edge filtering, k-NN, SVM

I. INTRODUCTION

In the era of increasing presence of electronic devices, there are still a lot of handwritten digitized documents. These documents can contain a lot of handwritten digits. Bank checks, postal ZIP codes on letters, everyday receipts, old medical records are some examples where handwritten digits prevail.

There are two types of handwritten character recognition methods, online and offline that are addressed differently. The online handwritten recognition methods need to learn only one person's writing pattern; the offline methods need to approach the way of writing for many people.

In offline handwritten digit recognition, the problems that arise are due to the fact that the same digit can be shaped very differently by different persons, there may be missing parts of the digit, the bias of the digit can vary a lot. Although we have a classification problem with only ten classes, the difficulty is the vast intra-class variability and one must build features with good inter-class separability.

A handwritten digit recognition system consists of several steps: preprocessing the image in order to make it easier to identify and extract the digits, another step is segmenting the image in order to identify and take out the digit, feature extraction which can be preceded by some transformations of the digit (deskewing, distortion), and classification. In this paper we only deal with the last two steps.

There are many methods for feature extraction, some of them rely on the shape of the digits, some take into account the topological structure, other use statistical properties of images with handwritten digits, or employ gradient or chain code extracted features [1], [2], [3], [4]. Our method for feature extraction uses directional information gathered from gradient filtered images.

In [5], [3] comparisons of classifiers for the handwritten digit problem are presented. The recent algorithms use combinations of features, or combination of classifiers or use deep learning techniques in order to improve the classification results [6]. In [7] the authors use six feature extraction methods: structural characteristics, modified edge maps, image projections, multi-zoning, concavity measurements, MAT-based gradient directional features with Multi-Layer Perceptron neural networks and combine the individual results in order to obtain 99.68% recognition rate on MNIST database [8], [9]. In [4] they use gradient, chain code and concavity features with k-NN and SVMs and test their idea on several digit datasets. The best result on MNIST is a 0.67% error rate with a combination of features and a Gaussian SVM. In [10] the author employs a special Artificial Neural Network called Restrictive Boltzman Machine in order to obtain a 99.22% recognition rate on MINST. In [11], [12], [13] the authors introduce a series of neural networks in order to obtain a 0.35% error rate on MINIST. The best classification result on MINIST (0.19% error rate) was obtained in [14] using a hybrid combination of a convolutional neural network that computes automatically features which are classified with a SVM.

II. MNIST DIGIT DATASET

The MNIST (Mixed National Institute of Standards and Technology) [8], [9] database contains handwritten digit images usually employed for testing digit recognition methods and, more generally, for machine learning algorithms.

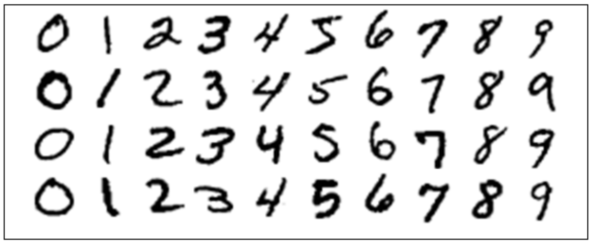
TABLE I. MNIST - NUMBER OF IMIGES FOR EACH DIGIT

Digit	Training	Test
0	5923	980
1	6742	1135
2	5958	1032
3	6131	1010
4	5842	982
5	5421	892
6	5918	958
7	6265	1028
8	5851	974
9	5949	1009

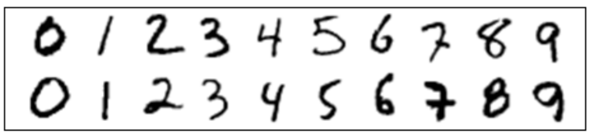
2470-881X/16 $31.00 © 2016 IEEE
DOI 10.1109/SYNASC.2016.48

The dataset has digit images of size 28×28, the background is black and the digits are grayscale (toward the white part of the grayscale interval). In this paper, we depicted the negative of the selected images. It has a training part consisting of 60000 digits and a 10000 digits' test part. In Table 1 are the number of images for each digit in MNIST.

In Fig. 1 are some samples from MNIST training and test datasets.



(a) – Digit samples from MNIST training set



(b) – Digit samples from MNIST test set

Figure 1. MNIST database handwritten digit examples

III. Feature Extraction Procedure

In order to extract features from images with digits we use simple image processing techniques such as image rotations, edge detection with Sobel filters, threshold binarization [15].

Rotations are geometrical transformations, the grid on which the image is represented being rotated with a certain angle and then the values of the pixels on the new grid are computed. There are three types of interpolation methods used in order to assign the new pixel's intensity: nearest neighbor, bilinear (using the values of the 8 nearest neighbors) and bicubic (uses 16 nearest values).

Sobel filters are 3×3 first order derivative filters employed to detect and emphasize lines in digital images. We use the following three types of Sobel filters: vertical, horizontal and diagonal (45º direction) edge detection kernels:

$$w_{vertical} = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}, \; w_{horizontal} = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \end{pmatrix}, \quad w_{diagonal} = \begin{pmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{pmatrix}. \qquad (1)$$

The image is convolved with the Sobel kernel to produce the filtered image where the desired direction is emphasized. In Fig. 2 the results of Sobel filtering an image are depicted.

Figure 2. Sobel filter results

We use these operations to extract features in the following way:

- first the image is rotated with angle $\boldsymbol{\theta}$,
- a Sobel filter is applied to the rotated image,
- the image is binarized using an appropriate threshold
- the binary image is divided in 4×4 blocks,
- for each block the number of black pixels are counted and stored in the feature vector.

For each rotation angle $\boldsymbol{\theta}$ one gets a 16-dimensional feature vector, $\boldsymbol{f_\theta}$.

As can be observed from the digit images, the middle part of the image contains more information than the peripheral parts. That's why the following Gaussian weights were employed for the feature vector:

$$G = \begin{pmatrix} 0.0298 & 0.0565 & 0.0565 & 0.0298 \\ 0.0565 & 0.1072 & 0.1072 & 0.0565 \\ 0.0565 & 0.1072 & 0.1072 & 0.0565 \\ 0.0298 & 0.0565 & 0.0565 & 0.0298 \end{pmatrix}. \qquad (2)$$

obtaining:

$$f_{\theta,G} = (g_{ij} f_k)\,, \;\; \text{where}\, f_\theta = (f_k)_{k=1}^{16}\,. \qquad (3)$$

The weights (2) were generated using a Gaussian distribution with zero mean and 1.25 standard deviation.

The final feature vector was built by concatenating feature vectors obtained using different sets of rotation angles:

$$f^c = \left(f_{\theta,G},\; \theta \in Angles\right). \qquad (4)$$

and then normalizing the result using the L_1 norm:

$$f = \left(\frac{f_i^c}{\sum f_i^c} \right) \qquad (5)$$

We also applied PCA (Principal Component Analysis) [16] to the final feature vectors (5) in order to reduce its size.

IV. RESULTS

The tests were performed using MATLAB. We used the following sets of angles:

1. $A_1 = \{0, 30º, 60º, 90º, \ldots, 300º, 330º\}$,
2. $A_2 = \{0, 10º, 40º, 70º, \ldots, 310º, 340º\}$,
3. $A_3 = \{0, 20º, 50º, 80º, \ldots, 320º, 350º\}$,
4. $A_4 = \{0, 20º, 40º, 60º, \ldots, 320º, 340º\}$,
5. $A_5 = \{0, 10º, 30º, 50º, \ldots, 330º, 350º\}$.

In Fig. 3 the images used in set A_3 are depicted. To these type of images, the Sobel filters are applied.

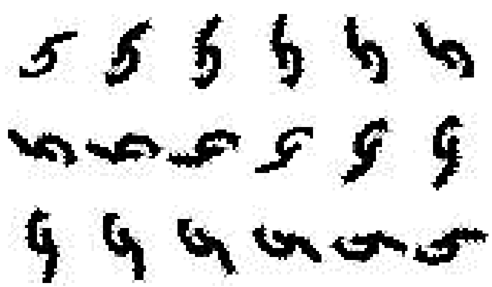


Figure 3. Rotated digit images for set A_3

The feature vectors have the following length: 192, 208, 208, 288, and 304, respectively. By using PCA in all cases the feature vectors were reduced to 150 dimension.

As classifiers k-NN with $k=3$ and Euclidean distance, and SVM with Gaussian kernel were employed [16], [17]. The Sobel filters were considered separately.

In Table 2 classification results are present for the five angles sets and three Sobel filters using 3-NN.

TABLE II. 3-NN CLASSIFICATION RESULTS

ANGLES SETS	TYPES OF SOBEL FILTERS		
	VERTICAL	HORIZONTAL	DIAGONAL
A_1	97.28%	97.27%	97.48%
A_2	97.35%	97.41%	97.60%
A_3	97.49%	97.68%	97.66%
A_4	97.54%	97.55%	97.94%
A_5	97.63%	97.64%	97.74%

In Table 3 are the same types of results as in Table 1 but in this case the classifier was a Gaussian SVM.

Note that for the k-NN classifier the results are above 97% and the best classification rate is obtained for the diagonal Sobel filter. When using SVMs the results improve with more the 1% and the best results usually are provided by the vertical Sobel filter.

In order to improve the classification results, for each of the five sets of angles, we combined the three results obtained with the SVM classifiers and different Sobel filters.

TABLE III. SVM CLASSIFICATION RESULTS

ANGLES SETS	TYPES OF SOBEL FILTERS		
	VERTICAL	HORIZONTAL	DIAGONAL
A_1	98.82%	98.73%	98.58%
A_2	98.76%	98.71%	98.64%
A_3	98.84%	98.84%	98.57%
A_4	98.96%	98.84%	98.85%
A_5	98.86%	98.88%	98.82%

The combination is a "majority voting" procedure. When two or all three classifiers agree on a label, then this is the label assigned to the tested image. For the SVM classifiers, the posterior probabilities that the analyzed image belongs to one of the ten digit classes were computed. When the classifiers point to three different digits two strategies were considered: first, the label of the best classifier was assigned and second, the average of the three posterior probabilities are computed and the class with maximal average probability was assigned.

The results of this type of combining the classifiers are in Table 4:

TABLE IV. COMBINING CLASSIFIERS RESULTS

ANGLES SET	COMBINING STRATEGY	
	'BEST'	AVERAGE
A_1	98.87%	98.88%
A_2	98.79%	98.79%
A_3	98.88%	98.91%
A_4	**99.05%**	99.04%
A_5	98.97%	98.97%

We get the best classification result (99.05%) for this kind of features for angles set A_4 by combining SVM results with three types of Sobel filtering, and choosing the best classifier label when the classifiers disagree. Note that combining the three classifiers yields always better results than the individual ones.

We also analyzed the classification rate for each digit separately. We show in Table 5 the results provided by the best k-NN, SVM and combined classifiers, i.e. for the angles set A_4. Observe that the best identified digit is 0, and the worst is 9. The identification order of the digits remains almost the same if we employ the other sets of angles, 9 is the worst identified digit and the best is either 0 or 1. Taking into account the method we used and the fact that 9 is an 180º rotated version of 6, we expected that digits 9 and 6 to have close classification rates, but that does not happen. This is due to the fact that in MNIST there are a lot of 9 digits that look like 4 and usually they are misidentified accordingly, which does not happen with the images of digit 6.

TABLE V. DIGIT CLASSIFICATION RESULTS

DIGIT	*k*NN	SVM	COMBINED
0	**99.49%**	**99.69%**	**99.80%**
1	99.12%	99.56%	99.56%
2	98.16%	99.03%	99.13%
3	97.43%	99.21%	99.21%
4	96.54%	99.08%	98.98%
5	96.86%	98.99%	98.99%
6	99.27%	99.06%	99.16%
7	97.37%	98.64%	98.74%
8	98.15%	98.87%	98.97%
9	**96.83%**	**97.42%**	**97.82%**

We tested our method on USPS, another very well-known digit database [18]. There are some differences between the results obtained with the two datasets. For USPS, a minimum distance classifier usually provided better results than 3-NN (with three exceptions). The A_4 set provided the best results for the SVM classifier with vertical or horizontal features and also for 3-NN with diagonal features. In the cases of a 3-NN classifier with vertical or horizontal features the best results were obtained with the A_3 set of angles and A_5 worked best for SVM with diagonal features. Contrary to the MNIST dataset, the "majority voting" procedure did not improve the individual results of the combined classifiers. The best identification rate (96.56%) was obtained for the SVM classifier with features provided by the horizontal Sobel filter and the A_4 set of angles, which for this database is a good result (2.5% error rate is considered "excellent" for this digit database). From the point of view of the classification rate for each digit, 0 has the best identification rate in all cases and 2 is the most misidentified digit. The 6 and 9 have similar classification results.

In order to improve our results, we need to focus on getting a better identification of the digits with lower recognition without losing the good results obtained with the other digits.

V. CONCLUSIONS

In this paper, we tested a new feature extraction method for handwritten digit recognition. The algorithm consists in repeatedly and thoroughly rotating the digit image and extract information provided by Sobel edge filtering the rotated image. We tested three types of edge filtering (vertical, horizontal, and 45º) using as classifiers *3*-NN and Gaussian SVMs. We also combined the classifiers in order to improve the results. Although our best result is not higher than those stated in the introduction, it is obtained only with one type of feature extraction procedure. It is possible that using deep learning techniques with this set of features to obtain better results.

We intend to find a combination of rotation angles that provide optimal results. Other edge filtering methods and other classifiers (AdaBoost, ANN) are to be tested in a future work.

REFERENCES

[1] I. Patel, V. Jagtap, O. Kale, "A Survey on Feature Extraction Methods for Handwritten Digits Recognition," Intl. J. on Computer Applications, vol. 107, no. 12, pp. 11-17, December 2014.

[2] P. Zhang, "Reliable recognition of handwritten digits using a cascade ensemble classifier system and hybrid features," PhD Thesis, Concordia University, Montreal, Quebec, Canada, 2006.

[3] Y. LeCun, L. D. Jackel, L. Bottou, A. Brunot, C. Cortes, J. S. Denker, H. Drucker,I. Guyon, U. Müller, E. Säckinger, P. Simard, and V. Vapnik, "Comparison of learning algorithms for handwritten digit recognition," *International Conference on Artificial Neural Networks* Vol. 60, pp. 53-60, October, 1995.

[4] M. Karic, and G. Martinovic, "Improving offline handwritten digit recognition using concavity-based features," *International Journal of Computers Communications & Control* 8.2, pp. 220-234, 2013.

[5] L. Bottou, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, L. D. Jackel, and V. Vapnik "Comparison of classifier methods: a case study in handwritten digit recognition," *International Conf. on Pattern Recognition*, pp. 77-77, IEEE Computer Society Press, October, 1994.

[6] L. Xu, A. Krzyzak, and Y. S. Ching, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Transactions on Systems, Man, and Cybernetics* 22.3, pp. 418-435, 1992.

[7] R. M. Cruz, G. D. Cavalcanti, and T.I. Ren, "Handwritten digit recognition using multiple feature extraction techniques and classifier ensemble", *17th International Conference on Systems, Signals and Image Processing*, pp. 215-218, 2010.

[8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 86(11), pp. 2278-2324, November 1998.

[9] MNIST database downloadable at http://yann.lecun.com/exdb/mnist/

[10] K. T. O'Shea, "Massively Deep Artificial Neural Networks for Handwritten Digit Recognition", *arXiv preprint arXiv:1507.05053*, July 2015.

[11] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Handwritten digit recognition with a committee of deep neural nets on gpus," *arXiv preprint arXiv:1103.4487*, 2011.

[12] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Deep Big Simple Neural Nets Excel on Handwritten Digit Recognition," *arXiv preprint arXiv:1003.0358*, March 2010.

[13] D. C. Cireşan, U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification," *arXiv preprint arXiv:1202.2745*, February 2012.

[14] X. X. Niu, and Y. S. Ching, "A novel hybrid CNN–SVM classifier for recognizing handwritten digits," *Pattern Recognition* 45.4 pp. 1318-1325, 2012.

[15] R. C. Gonzalez, and R. E. Woods, Digital Image Processing, 3rd ed., Prentice Hall, Upper Saddle River, NJ, 2008.

[16] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[17] S. Theodoridis and K. Koutroumbas, Pattern Recognition, 4th ed., Academic Press, 2009.

[18] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/multiclass/ - USPS database.