

Handwritten Digit Recognition using Slope Detail Features

A. M. Hafiz
Department of Electronics
University of Kashmir, J and K, India

G. M. Bhat
Department of Electronics
University of Kashmir, J and K, India

ABSTRACT

In this paper, new features called Slope Detail (SD) features for handwritten digit recognition have been introduced. These features are based on shape analysis of the digit image and extract slant or slope information. They are effective in obtaining good recognition accuracies. When combined with commonly used features, Slope Detail features enhance the digit recognition accuracy. K- Nearest Neighbour (k-NN) and Support Vector Machine (SVM) algorithms have been used for classification purposes. The data sets used are the Semeion Data Set and United States Postal Service (USPS) Data Set. For the USPS Data Set an error rate of 1.3% was obtained, which has been found to be better than any reported error rate on the said data set.

General Terms

Histogram Features, OCR, Pattern Recognition, Artificial Intelligence.

Keywords

Slope Features; Handwritten digits; Pattern Classification; Nearest Neighbor; Support Vector Machine; Artificial Intelligence; Gradient Feature; USPS Data Set;

1. INTRODUCTION

Handwritten digit recognition is an area of active research for applications like recognizing zip code in postal offices (offline recognition) and recognition of digits on handheld devices like mobile phones, tablets, etc. (online recognition). Algorithms like K-Nearest Neighbour (K-NN) and Support Vector Machine (SVM) have been commonly used for digit recognition.

Written text recognition encompasses automated recognition of both offline as well as online text [1]. The number of classes varies with the language. For example, there are over 7000 character classes for Chinese text alone [2]. Various classifiers have been used for this purpose which include K-NN [3], SVM [3], Hidden Markov Model (HMM) [4], Artificial Neural Network (ANN) [5], Modified Quadratic Discriminant Function (MQDF) [2], etc.

Handwritten digit recognition has developed into an interesting area of contemporary research. While it primarily remains a classification problem, feature extraction methods have been used by many researchers. Conventional features like Gradient features have been used for the purpose. An attempt is made to enhance the digit recognition accuracy by incorporating a new feature called Slope Detail (SD) Feature of the handwritten digit. The experimental observation has shown an improvement in the accuracy of the recognition. Semeion Handwritten Digit (SHD) Data Set [6] and United States Postal Service (USPS) Data Set [7] have been used for

experimentation purposes. SHD Data Set, a relatively difficult dataset, is available on UCI Machine Learning Repository [8]. The SHD Data Set has 1593 scanned digits from around 80 persons. Out of these 1000 samples (100x10) have been used for training and 500 samples (50x10) have been used for testing. Another popular Data set called the USPS Data Set available online, has also been used for experimentation purposes [7]. It consists of 1100 instances of each digit (0-9). It is a relatively difficult dataset for classification purposes. However, due to its medium size, it is a popular choice for researchers. In this work, the first 900 instances of each digit were used for training and the remaining 200 instances were used for testing. Thus, training set size was 9000 and testing set size was 2000.

In the first part of the paper, an overview of K-NN and SVM algorithms has been given. In the second part, the conventional feature extraction based on Histogram and Gradient features has been implemented. New features called Slope Detail (SD) Features, which are extracted from the given data sets have been presented. Extensive experimental analysis of handwritten digit recognition using the proposed features has been discussed in the third part of this paper. It can be observed that, SD Features when combined with other available features improve the classification accuracy.

2. CLASSIFICATION ALGORITHMS – SVM AND K-NN

2.1 Support Vector Machine

The goal of SVM is to produce a model (based on the training data) which predicts the target values of the test data given only the test data attributes [9]. Given a training set of instance-label pairs (\mathbf{x}_i, y_i) , $i = 1, \dots, l$, $\mathbf{x}_i \in R^n$, $y_i \in [1, -1]^l$, the support vector machines (SVM), [10], require the solution of the following optimization problem:

$$\min_{\mathbf{w}, b, \xi} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^l \xi_i$$

subject to

$$y_i (\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \quad (1)$$
$$\xi_i \geq 0.$$

Radial Basis Function (RBF) [11] was used for SVM in the experiments:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2), \quad \gamma > 0 \quad (2)$$

Here γ is a kernel parameter. LIBSVM [12] was used for training and testing the datasets by SVM method. For optimizing SVM performance, best possible combination of

' γ ' and a penalty factor 'C' has to be used. To find these an iterative algorithm with initial coarse search and subsequent fine search was used. 5-folds cross-validation was used to find optimum values for C and γ .

2.2 k-Nearest Neighbour

The approach was proposed by T.M. Cover and P.E. Hart in 1967 [13]. In K-NN, the distances between feature vector to be classified (and all feature vectors in the dataset) are computed using a distance metric. The test feature vector is said to belong to the class of feature vector(s) with minimum distance. A set of n pairs $(x_1, t_1), (x_2, t_2), \dots, (x_n, t_n)$ is given, where the x_i 's take values in a metric space X upon which is defined a metric d , and the t_i 's take values in the set $[1, 2, \dots, M]$. Each t_i is considered to be the index of the category to which the i^{th} individual belongs, and each x_i is the outcome of the set of measurements made upon that individual. A new pair (x', t) is given, and it is desired to estimate t . $x_n \in \{x_1, x_2, \dots, x_n\}$ is the nearest neighbor to x' if

$$\min d(x_i, x') = d(x'_n, x') \quad i = 1, 2, \dots, n \quad (3)$$

This classification is for 1-NN. For k-NN, the classifier finds the k nearest neighbors of x' and then predicts t as the most frequent one occurring in the K neighbors [14]. The metric used was the City-block (CB) Distance [15]:

$$d_{cb} = \sum_i |x_i - x'_i| \quad (4)$$

3. FEATURES FOR HANDWRITTEN DIGIT RECOGNITION

Depending on the features given to the classifier, it accumulates a knowledge base for classification purposes. In case of a binary image, when translated into an array and used as an attribute, no information is given to the classifier about the order of the attributes. It would in fact be irrelevant if all the patterns are shuffled in the same manner and presented to the classifier for classification purposes. If such information is given to the classifier, its performance can be improved. One such feature (Pixel Count Feature) is obtained by counting row-wise, number of black pixels present and doing same column-wise, thus obtaining two profiles. For example the row profile \mathbf{r} of dimensions $(1 \times N)$ can be obtained from complemented binary image I of $N \times N$ pixels, where 0s and 1s represent white and black pixels, respectively, by the following operation:

$$\mathbf{r}_i = \sum_{j=1}^N I(j, i) \quad i = 1, \dots, N \quad (5)$$

These are shown in Fig. 1.



Fig. 1. Row and Column Pixel Count (PC) Features

Two more features added to this set are variance of number of black pixels in rows and variance of number of black pixels in columns. Another useful feature is the character shape profile, where distance in pixels from an edge of the image to a black

pixel is found. This is done from all four edges, giving four profiles. These are shown in Fig. 2.

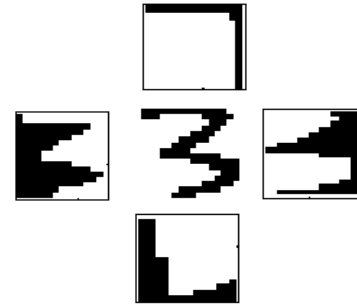


Fig. 2.Character Profile (CP) Features

If the difference of the $(n + 1)^{\text{th}}$ and n^{th} elements in each CP profile is taken, a new feature is obtained. This is done as:

$$f'_i = f_{i+1} - f_i \quad i = 1, \dots, N - 1 \quad (6)$$

where $N = \text{Length of CP Profile}$. This is a new type of information and is shown in Fig. 3.

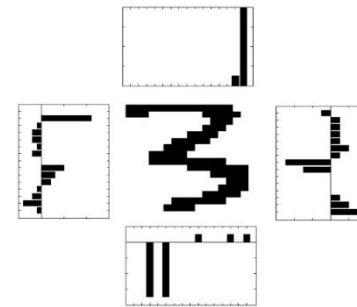


Fig. 3.Character Profile Difference (CPD) Features

Here, a lot of feature variation is observed even for similar patterns, which leads to lesser gain in classification accuracy. If only the sign of this difference $[-, 0, +]$ is taken into consideration, a superior feature is obtained, with a considerable gain in classification accuracy. For two successive pixels in a character profile, a southwest slant is given value 1, a southeast slant is given value 3, and a straight vertical or horizontal line is given value 2. This can be represented as:

$$f''_i = \begin{cases} 1, & \text{if } f'_i < 0 \\ 2, & \text{if } f'_i = 0 \\ 3, & \text{if } f'_i > 0 \end{cases} \quad (7)$$

Thus the minor irregularities even in case of patterns belonging to same class are avoided. Since this feature gives information about the slant or slope of the profile viz. positive, negative or zero, it is called Slope Detail (SD) Feature. For the same digit, the Plain Slope Detail (PSD) features are shown in Fig. 4.

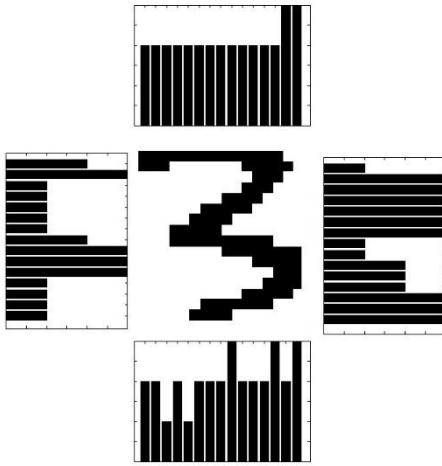


Fig. 4. PSD Features

More information about a discontinuity (empty space between 2 black pixels) in a PSD Profile is added by replacing the attribute value by zero. This is done by running an algorithm. The modified profile is the SD Profile and this is the SD Feature.

For comparing performance of SD Features, Gradient Features [16] have been used. The latter are robust and popular features used for OCR. To extract the Gradient Features, firstly, Sobel operators were used for finding gradients of strokes in horizontal and vertical directions, respectively, in the binary image. The horizontal Sobel operator used was:

$$Grd_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

The vertical Sobel operator used was:

$$Grd_y = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Subsequently, the magnitude and phase of the gradient was found at each pixel as:

$$M_i = \sqrt{x_i^2 + y_i^2} \quad (8)$$

$$P_i = \tan^{-1} \left(\frac{y_i}{x_i} \right) \quad (9)$$

where M_i and P_i are magnitude and phase of gradient at i^{th} pixel in binary image respectively ($i = 1, 2, 3, \dots, 256$). Also, x_i and y_i are horizontal and vertical components of the gradient at i^{th} pixel after taking respective Sobel operators. Next 4 axis or 8 directions were assigned in plane of image, where every 2 neighboring axis/directions are separated by an angle of 45° . Thus, the directions were straight lines at following angles (taken anticlockwise): 0, 45, 90, 135, 180, 225, 270, 315 degrees. The components of the vector $M_i \angle P_i$ along its two neighboring direction vectors were found. For all possible variations, 8 features per pixel in binary image (for 8 directions) were obtained. Thus the gradient components of the binary image were reduced to 8 gradient planes. Then the image was divided into 16 equally sized blocks i.e. 4 horizontal blocks and 4 vertical blocks. Then the gradient components along 8 respective planes for each block of the image were added. This resulted in an $8 \times 16 = 128$ element feature per image.

4. TESTING

Classification was done using both k-NN and SVM. Features like SD Feature (or s), PSD Feature (or s^j), Grayscale image pixel values (or r), Binary image pixel values (or b), PC Feature (or h), CP Feature (or a or c) and Gradient Feature (or g) were used alone or in combination with each other. Certain features were duplicated in feature set for optimizing classification accuracy. This was not found to have any adverse effects as concluded from cross-validation. Classification was done using k-NN. Classification on the feature sets (given below) was also done using SVM from LIBSVM Package for Matlab. The following observations were made (as shown in tables) for feature combinations showing significant classification accuracy. Feature re-use has been practiced, which yields good results as is observed.

Table 1. Results for USPS Data Set Classification using SVM (with Feature Re-use)

Features	Accuracy (%)
r	97.85
b	97.95
bsh	98.15
rsh	98.35
rshh	98.50
rshhh	98.60
rshhhh	98.70*
rshhhhh	98.60
sh	97.40
shh	97.55
g	98.40
gh	98.55
ghr	98.55
ghhr	98.60*
ghhhr	98.55

Table 2. Results for USPS Data Set Classification using k-NN

Features	Accuracy (%)
b	96.65
bch	97.15
bsh	97.35*
g	96.75
gh	97.30*

Table 3. Results for Semeion Data Set Classification using SVM

Features	Accuracy (%)
G	97.00
Gh	96.00
Ghr	96.00
Gha	97.40*
Rhs	96.20
Rhsa	95.80
rhss ¹	96.40
hss ¹	97.80*

Table 4. Results for Semeion Data Set Classification using k-NN

Features	Accuracy (%)
G	94.00
Gc	95.40
Gh	95.60
ghc	95.80
ghcc	95.80
ghcr	95.60
ghcs	96.00*
Rs	94.20
Rh	94.20
Rch	95.40
Rsh	95.60
rhss ¹	96.80*
rhs ¹	95.40
hss ¹	96.00
Rs	94.8

Dimensionally reduced USPS testing datasets for best feature combinations (for SVM) using SD and Gradient features respectively are shown in Fig. 5.

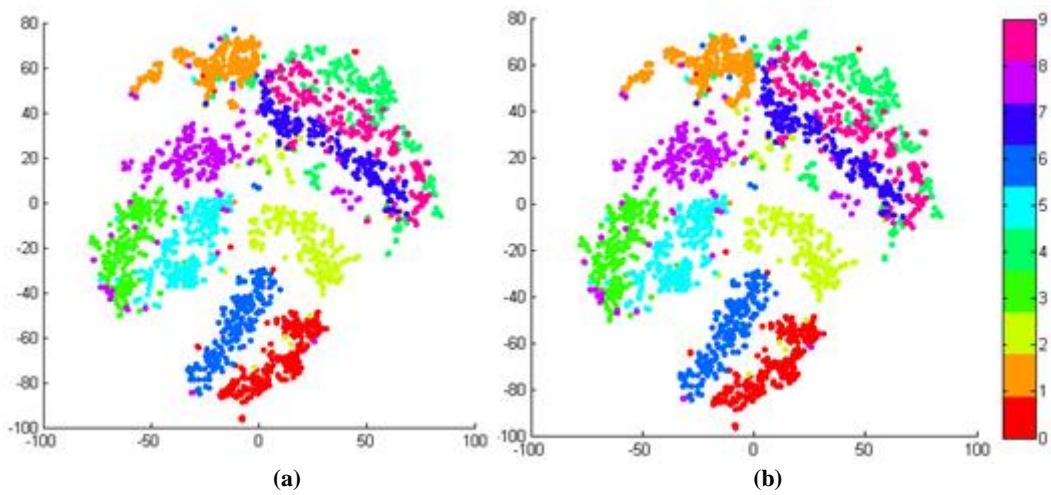


Fig. 5 (a) rshhhh feature-combination, (b) ghhr feature-combination

Dimensional Reduction has been done using t-SNE [17] from 452 initial dimensions to 2 dimensions. Better cluster separation (for digits ‘1’ and ‘4’) is observed for SD Features as compared to that of Gradient Features.

From our experiments it was observed that the highest classification accuracy of 98.70% was obtained for SD features for USPS while as gradient feature gave an accuracy of 98.60% when SVM was used. Similarly, using SVM

method for classifying Semeion data set, highest accuracy of 97.80% was obtained using SD features while as Gradient features yielded a rate of 97.40%.

The k-NN method also yielded higher accuracy for SD features than that of Gradient features for both datasets. A comparison of prominent research on USPS classification is shown in Table 5.

Table 5. Performance Comparison for USPS Dataset (USPS+ refers to extended dataset)

Classifiers	Training set	Error Rate %
Nearest-neighbor [18]	USPS+	5.9
LeNet1[19]	USPS+	5.0
Optimal margin Classifier [20]	USPS	4.6
SVM [21]	USPS	4.0
Latent Log-Linear Models [22]	USPS	3.7
Local Learning [23]	USPS+	3.3
Virtual SVM [24]	USPS	3.2
Virtual SVM, local kernel [25]	USPS	3.0
Transformation Knowledge with Kernel Methods [26]	USPS	2.8
Tangent Distance [18]	USPS+	2.6
Deformation Models [27]	USPS	2.4
Feature-based virtual SVM [28]	USPS	2.34
SMO [29]	USPS	2.24
Combination of tangent vector and local representation [30]	USPS	2.0
Proposed SD Features	USPS+	1.30*

A comparison of various digit recognition techniques in Table 5 shows that the Proposed SD Features have a significant advantage. Previous reported work indicates the best recognition error rate of 2.0%. It has been observed in [30] that the improvement was obtained by training on the testing data as well, which is uncommon. It also violates the fundamental rule of pattern recognition that testing subset should be hidden while training is going on for the classifier. The proposed technique does not suffer from such shortcomings, and can, thus, serve as an efficient digit recognition technique.

5. CONCLUSION

In this paper various features used for handwritten digit recognition have been discussed. New digit recognition features called Slope Detail (SD) features have been introduced. The efficiency of digit recognition using SD features for classification has been compared with that of other conventional features. For this purpose K-NN and SVM classifiers have been used on the Semeion data set and popular USPS data set, both relatively difficult datasets. The experimental results show that the classification accuracy is improved when SD Features are used in combination with other features. It further enhances the classification performance for USPS dataset.

6. REFERENCES

- [1] R. Plamondon and S. N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 68-69, 2000.
- [2] Q. F. Wang, F. Yin, and C. L. Liu, "Handwritten Chinese Text Recognition by Integrating Multiple Contexts," *Trans. Pattern Analysis and Machine Intelligence*, vol. 34, 2012.
- [3] A. M. Namboodiri and A. K. Jain, "Online Handwritten Script Recognition," *Trans. Pattern Analysis and Machine Intelligence*, vol. 26, 2004.
- [4] A. Vinciarelli, S. Bengio, and H. Bunke, "Offline Recognition of Unconstrained Handwritten Texts Using HMMs and Statistical Language Models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, 2004.
- [5] S. Marinai, M. Gori, and G. Soda, "Artificial Neural Networks for Document Analysis and Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, 2005.
- [6] ed. Semeion Research Center of Sciences of Communication, via Sersale 117, 00128 Rome, Italy, Tattile Via Gaetano Donizetti, 1-3-5, 25030 Mairano (Brescia), Italy.
- [7] S. Roweis. USPS Handwritten Digit Dataset [Online]. Available: <http://www.cs.nyu.edu/~roweis/data.html>
- [8] A. Frank and A. Asuncion, ed. UCI Machine Learning Repository, School of Information and Computer Science, University of California, Irvine, CA, 2010.
- [9] C. C. Chang, C. W. Hsu, and C. J. Lin. (2009, Practical Guide to Support Vector Classification.
- [10] C. Cortes and V. Vapnik, "Support-vector network," *Machine Learning*, vol. 20, pp. 273-297, 1995.
- [11] S. S. Keerthi and C. J. Lin, "Asymptotic behaviors of support vector machines with gaussian kernel," *Neural Computation*, 2003.
- [12] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 1-27, 2011.
- [13] T. M. Cover and P. E. Hart, "Nearest Neighbor Pattern Classification," *IEEE Transactions on Information Theory*, vol. 13, pp. 21-27, 1967.
- [14] P. Cunningham and S. J. Delaney. (2007, k-Nearest Neighbour Classifiers.
- [15] M. Weeks, V. Hodge, S. O'Keefe, J. Austin, and K. Lees, "Improved AURA k-nearest neighbour approach," in *Proc. of the 7th International Work-Conference on Artificial and Natural Neural Networks*, J. Mira and J. R. Alvarez, Eds., ed, 2003.
- [16] H. Liu and X. Ding, "Handwritten Character Recognition Using Gradient Feature and Quadratic Classifier with Multiple Discrimination Schemes," in *8th ICDAR*, 2005, pp. 19-23.
- [17] L. J. P. v. d. Maaten and G. E. Hinton, "Visualizing High-Dimensional Data Using t-SNE," *Journal of*

Machine Learning Research, vol. 9, pp. 2579-2605, 2008.

- [18] P. Simard, Y. LeCun, and J. Denker, "Tangent prop - a formalism for specifying selected invariance in an adaptive network," in *Advances in Neural Information Processing Systems*. vol. 4, J. E. Moody, S. J. Hanson, and R. P. Lippmann, Eds., ed San Mateo, CA: Morgan Kaufmann, 1993.
- [19] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, pp. 541-551, 1989.
- [20] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *5th Annual ACM Workshop on Computational Learning Theory*, Pittsburg, PA, 1992.
- [21] B. Scholkopf, C. J. C. Burges, and V. Vapnik, "Extracting support data for a given task," in *First International Conference on Knowledge Discovery and Data Mining*, 1995, pp. 252-257.
- [22] T. Deselaers, T. Gass, G. Heigold, and H. Ney, "Latent Log-Linear Models for Handwritten Digit Classification," vol. 34, pp. 1105 - 1117, 2012.
- [23] L. Bottou and V. Vapnik, "Local Learning algorithm," *Neural Computation*, vol. 4, pp. 888-901, 1992.
- [24] B. Scholkopf, C. J. C. Burges, and V. Vapnik, "Incorporating invariances in support vector learning machines," in *Artificial Neural Network*, 1996, pp. 47-52.
- [25] B. Scholkopf, "Support Vector Learning," Ph.D., Technical University of Berlin, Munchen, 1997.
- [26] B. Haasdonk, "Transformation Knowledge in Pattern Analysis with Kernel Methods," Ph.D Ph.D., Albert-Ludwigs-Universitat, Freiburg, 2005.
- [27] D. Keysers, T. Deselaers, C. Gollan, and H. Ney, "Deformation models for image recognition," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 1422-1435, 2007.
- [28] J. X. Dong, A. Krzyzak, and C. Y. Suen, "Fast SVM Training Algorithm with Decomposition on Very Large Datasets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 603-618, 2005.
- [29] J.-x. Dong, A. Krzyzak, and C. Y. Suen, "A Practical SMO Algorithm for Pattern Classification," 2001.
- [30] D. Keysers, R. Paredes, H. Ney, and E. Vidal, "Combination of Tangent Vectors and Local Representation for Handwritten Digit Recognition," in *International Workshop on Statistical Pattern Recognition*, 2002, pp. 538-547.
- [31] J. Bromley and E. Sackinger, "Neural-network and k-nearest-neighbor classifiers," AT&T1991.
- [32] J.-x. Dong, "Statistical Results of Human Performance on USPS database," Concordia University 2001.