SVM Based Real Time Hand-Written Digit Recognition System

Conference Paper · January 2019

CITATIONS
0

READS
121

3 authors, including:

Hafiz Ahamed
Rajshahi University of Engineering & Technology
5 PUBLICATIONS **4** CITATIONS

SEE PROFILE

SVM Based Real Time Hand-Written Digit Recognition System

Hafiz Ahamed*, Ishraq Alam*, Md. Manirul Islam

Department of Mechatronics Engineering

Rajshahi University of Engineering & Technology

Rajshahi, Bangladesh

hafiz.mte13.ruet@gmail.com, ishraqmte.ruet@gmail.com, mmaniruleee@gmail.com

*Corresponding author [Hafiz Ahamed, Ishraq Alam]

Keywords:

- SVM;
- Handwritten digit;
- Real Time Recognition:

Abstract: Meanwhile Neural Networks based algorithms have intimated steadfast potential on various visual tasks including the recognition of Digits. This paper presents Support Vector Machine (SVM) based Real Time Hand-Written Digit Recognition System. The system involves two main sections i.e. training and recognition section. SVM classifier is used as the training algorithm and then tested it on MNIST dataset. We achieved a training accuracy of 98.05% and a test accuracy of 97.83% demonstrating that the proposed method can achieve significant and promising performance in digit recognition. Then we implemented our model to recognize user given handwritten digits in real time.

1. Introduction

Digit recognition is the widely studied part of character recognition which is most enthralling part of Computer Vision and Robotics. The most common application of digit recognition are license plate recognition [12], in banks for credit card number recognition [9], street number recognition, in post offices for sorting the mail, phone number recognition [7] etc. As posit, handwritten digit, printed digit and typewritten digit are three classification of digit found in application [13] Among them for any digit recognition system, printed and typewritten digit are more predictable. They can be recognized in a more precise manner deeming to the fact representing the combination of a finite number of different characters. Variations of digits depending on the font change are simpler for recognition. Every font preserves basic structure of then digit. So, any digit can be found in various different forms. While handwritten digit may have infinite number of forms depending on the handwriting of the person. Hand-written digit recognition is much more difficult compared to printed and typewritten digit. In this paper we proposed Support Vector Machine (SVM) as the Neural network-based classification techniques have been widely used in recent time for character recognition.

In very recent times Milad Mozafari et al. [10] presented a DCSNN for digit recognition which is trained by a combination of STDP and R-STDP learning rules. Their network achieved 97.2% of recognition accuracy. Jithendar Anumula et al. [2] presented a system which classifies a spoken digit based on the output spikes of the DAS in response to digits spoken into the on-board

microphones with an accuracy of 82%. Junfei Qiao et al. [11] proposed a Q-ADBN for handwritten digits recognition which is is inspired by deep reinforcement learning strategy, which performs well in terms of feature extracting and decision-making. Q-learning algorithm is used in decisionmaking in the proposed model. Abdel Jalil Gattal et al. [5] proposed a segmentation and recognition system for unknownlength handwritten digit strings by combining several explicit segmentation methods depending on the configuration link between digits. Three segmentation methods were combined based on histogram of the vertical projection, the contour analysis and the sliding window Randon transform.

In this paper, our main focus was to recognize hand written digit in real time. The digit recognition task followed by several pre-processing task and classifier training task. SVM was used as the classifier model to recognize the digits. The rest of the paper is arranged as follows: Methodology is described in section 2. Experimental results & discussions are shown in section 3. We conclude our paper in section 4.

2. Methodology

This portion deals with acquisition, some pre-processing, thresholding and feature extraction process of input image consist of user's hand-written digit. This steps are specified as follows:

2.1 Image Acquisition

Acquisition task is considered as the first step of user provided hand written digit recognition. Image acquisition process is mainly concern with obtaining text image from a scanner or a pre-stored image file where the format of the image may be PNG, BMP, JPEG, etc. A digital camera, scanner, webcam or any input devices is used to capture the input image. To make the image as readable format some preprocessing and thresholding task is performed on that image.

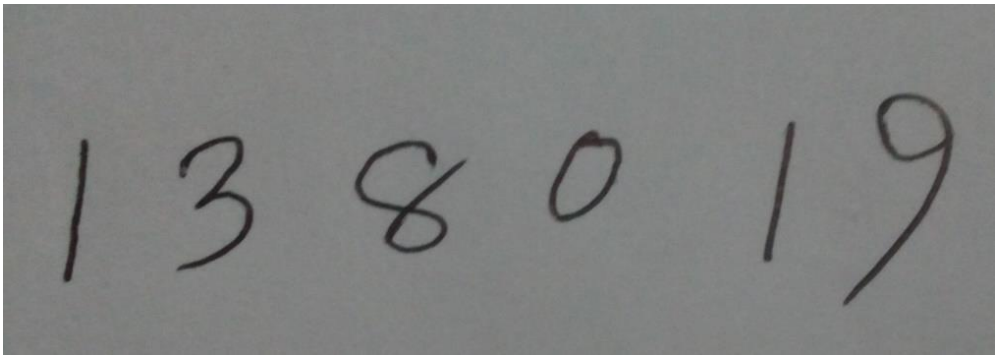


Fig. 1 Input Image

2.2 Pre-Processing

The acquired input image is subjected to a number of prefatory processing steps to make it functional to the descriptive stages of recognition process. The intention of Pre-processing is to produce data that are readable for the recognition systems to operate precisely. Pre-processing steps are as follows:

2.2.1 Gray Image Conversion

Grayscale images consist of many shades of gray. It is a result of calculating individual pixel intensity of the acquired image where range of lower pixel value represents the black pixel intensity and higher value represents white pixel intensity. For achieving accuracy of input document a RGB image is transformed into gray image which consist of pixel value from 0 to 255. The transformation [1] process from RGB to gray is followed by the equation

$$Y(i, j) = 0.299 \times R(i, j) + 0.587 \times G(i, j) + 0.114 \times B(i, j) \quad (1)$$

Where *R(i; j), G(i; j)* & *B(i; j)* represents the corresponding Red, Green and Blue color element of the image and *Y (i; j)* is the gray image.

2.2.2 Binarization & Thresholding

At the former step the image is converted RGB into gray image which is required to convert into a binary image for further processes of analysis and recognition. Binarization process involves in separating the pixel values of the gray scale image into two sessions 0 and 255 according to the threshold value. Thresholding process involves the setting of the contextual values for pixels beneath a threshold value and assigning an expected dissimilar value for the pixels. To get the featured object as white image and the background as black image we need to perform the inverse thresholding [1] process followed by the equation:

$$Y(i, j) = \begin{cases} 0 & if\ Y(i, j) > T \\ 255 & Otherwise \end{cases} \quad (2)$$

Here *T* represents the threshold number.

Fig. 2 Input Image after performing Inverse Thresholding operation.

2.3 Feature Extraction

Feature extraction process involves of extracting a set of features and differentiating the object features from raw background. This process provides the featured information about an object or a group of objects in order to assist classification task. After performing the feature extraction task, the features are represented as a feature vector. n order to avoid the additional complexity and to improve the accuracy of the classifier, an additional compressed and distinctive representation is obligatory. The digit which has to be recognized should be white having the pixel value 255 and the background should be black having the pixel value 0. Binary Inverse

Thresholding was used to threshold the image to find the needed pixel value. Contour [6] search technique was used for feature extraction where contour can be defined simply as a curve joining all the continuous points having same color or intensity. In many applications such as shape analysis, object detection and recognition contour can be used as an useful means.

2.4 Training Algorithm

The main decision-making task of recognizing a digit is performed by the classifier model. Using the extracted features followed by previous stages identify the digits. In this work SVM a machine learning algorithm is proposed as a classifier model.

Support Vector Machine (SVM) a binary classifier creates a hyper plane or group of hyper planes in a high or infinite dimensional space for separating data from different classes. A set of labeled training pattern $(y_1, x_1)\ldots\ldots(y_M, x_M)$ where, $y_i \in \{-1,1\}$ is said to be linearly separable if there exists a vector w and a scalar b such that the inequalities [3]

$$w.\, x_i + b \geq 1 \quad \text{if } y_i = 1 \tag{3}$$

$$w.\, x_i + b \leq 1 \quad \text{if } y_i = -1 \tag{4}$$

are valid for all elements of the training set. Thus the optimal hyperplane is

$$w_0\, x + b_0 = 0 \tag{5}$$

the unique one which separates the training data with a maximal margin.

Hyperplane should be as far as possible from instances of both classes. The distance that should be maximized is $\rho_0 = \frac{2}{\|w\|}$. As real time data is noise contaminated thus, we can't separate training data without error. In this case the training set has to be separated with a minimal number of errors. So soft margin was proposed to solve this problem. A slack variable is introduced to allow some instances to be misclassified. The definition is as follows:

$$y_i(w.\, x_i + b) \geq 1 - \epsilon_i \;;\; \epsilon_i \geq 0\,, 1 \leq i \leq n \tag{6}$$

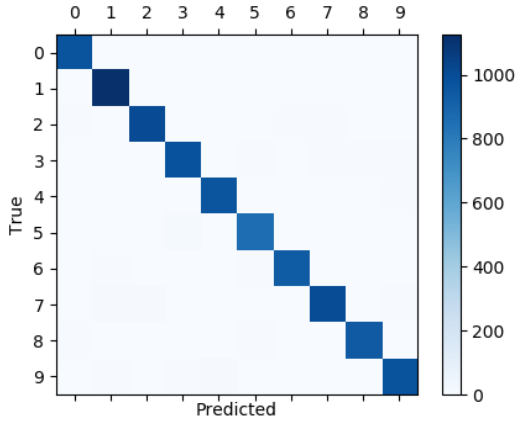
A kernel function is used instead of dot product to make SVM suitable for nonlinearly separable data. In practice Gaussian Radial Basis Function (RBF), Polynomial function and Sigmoid function are used as kernel function. In this proposed work Polynomial function is used as the kernel function which is only dependable on the number of support vectors not on the degree of polynomial [8].

$$K(x_i,\ x_j) = (1 + x_j{}^T x_i)^d \tag{7}$$

Where d represents Degree of Polynomial and K is the Kernel function.

3. Results & Discussion

The developed algorithm was designed and implemented on python *3.6*. First, we used the proposed algorithm on MNIST Dataset. General site for the MNIST [4] database: http://yann.lecun.com/exdb/mnist. The MNIST database was constructed out of the original NIST database; hence modified NIST or MNIST. There are *60,000* training images and *10,000* test images, both drawn from the same distribution. The Dimensionality of each image sample vector is *28×28 = 784*, where each element is binary. In training phase, we trained the SVM classifier based on MNIST database and then we tested our algorithm on MNIST test dataset. The training and test set accuracy was 98.05% and 97.83% respectively. respectively. Finally, we plotted the confusion matrix to visualize the accuracy corresponding to each of the input and output classes.




```
[[ 972    0    1    1    0    3    1    0    2    0]
 [   0 1124    2    2    0    0    3    0    4    0]
 [   7    0 1004    1    2    1    5    9    3    0]
 [   0    2    1  989    0    5    0    3    5    5]
 [   3    0    2    0  962    0    3    1    0   11]
 [   3    0    1   13    1  861    4    1    5    3]
 [   5    5    1    0    3    5  937    0    2    0]
 [   0   10    8    1    1    0    0 1002    0    6]
 [   5    0    1    2    5    4    0    3  952    2]
 [   3    6    2    5   11    3    1    3    2  973]]
```

Fig. 3 Confusion Matrix of Support Vector Machine.

Figure 3 represents the confusion matrix which is generated from SVM. It shows the number of test set data which are classified and misclassified respectively.

After performing the training phase, we implemented our trained model in our previously acquired input image containing hand written digit. The trained model recognized the digit and displayed the output drawing a bounding box containing each digit then write the corresponding recognized digit as text on the bounding box. Figure 4 represents the digit recognition in real time where every digit is recognized successfully.

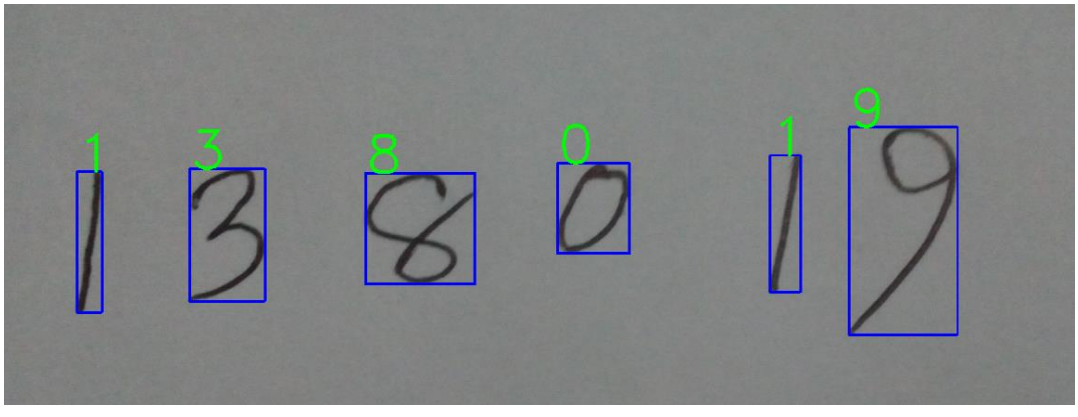


Fig. 4 Digit Recognition in real time.

4. Conclusion

In this proposed work, we present an employment of automatic recognition of handwritten digits implemented based on a noble classifier algorithm Support Vector Machine. When it is the fact of taking real time input data(image) the process always starts with noise removal and some preprocessing task. If we don't process the input picture properly classifier can't classify the digit appropriately as there is a chance of missing some vital information from feature vectors. At the training phase SVM classifier was trained based on MNIST dataset and the classifier performance was evaluated which provided a satisfactory result. After that the trained model was implemented on the acquired image for performing the digit recognition task.

5. References

[1] Shyla Afroge, Boshir Ahmed, and Firoz Mahmud. "Optical character recognition using back propagation neural network". In: Electrical, Computer & Telecommunication Engineering (ICECTE), International Conference on. IEEE. 2016, pp. 1–4.

[2] Jithendar Anumula et al. "Live demonstration: event driven real-time spoken digit recognition system". In: Circuits and Systems (ISCAS), 2017 IEEE International Symposium on. IEEE. 2017, pp. 1–1.

[3] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: Machine learning 20.3 (1995), pp. 273– 297.

[4] Li Deng. "The MNIST database of handwritten digit images for machine learning research [best of the web]". In: IEEE Signal Processing Magazine 29.6 (2012), pp. 141–142.

[5] Abdeljalil Gattal, Youcef Chibani, and Bilal Hadjadji. "Segmentation and recognition system for unknownlength handwritten digit strings". In: Pattern Analysis and Applications 20.2 (2017), pp. 307 323.

[6] Ruchi Manish Gurav and Premanand K Kadbe. "Real time finger tracking and contour detection for gesture recognition using OpenCV". In: Industrial Instrumentation and Control (ICIC), 2015 International Conference on. IEEE. 2015, pp. 974–977.

[7] Peter A Kalmstrom. Phone number recognition. US Patent 9,955,019. Apr. 2018.

[8] Kuo-Ping Lin et al. "Revenue forecasting using a leastsquares support vector regression model in a fuzzy environment". In: Information Sciences 220 (2013), pp. 196–209.

[9] Li Liu, Linlin Huang, and Jian Xue. "Identification of serial number on bank card using recurrent neural network". In: Ninth International Conference on Graphic and Image Processing (ICGIP 2017). Vol. 10615. International Society for Optics and Photonics. 2018, p. 1061516.

[10] Milad Mozafari et al. "Combining STDP and RewardModulated STDP in Deep Convolutional Spiking Neural Networks for Digit Recognition". In: arXiv preprint arXiv:1804.00227 (2018).

[11] Junfei Qiao et al. "An adaptive deep Q-learning strategy for handwritten digit recognition". In: Neural Networks (2018).

[12] Muhammad Sarfraz and Mohammed Jameel Ahmed. "An Approach to License Plate Recognition System Using Neural Network". In: Exploring Critical Approaches of Evolutionary Computation. IGI Global, 2019, pp. 20–36.

[13] Dipti Singh et al. "An application of SVM in character recognition with chain code". In: Communication, Control and Intelligent Systems (CCIS), 2015. IEEE. 2015, pp. 167–171.