Handwritten Character Recognition

Kiran B. Ingale
Department of Electronics and telecommunication
Vishwakarma Institute of Technology
Pune, Maharashtra, India

Akshay S. Mane
Department of Electronics and telecommunication
Vishwakarma Institute of Technology
Pune, Maharashtra, India

Vishwajeet V. Swami
Department of Electronics and telecommunication
Vishwakarma Institute of Technology
Pune, Maharashtra, India

***Abstract*— Handwritten digit and letter recognition is a very important topic in the field of image processing and pattern recognition. Handwritten digit and letter recognition poses different problem because of different writing styles, similarity in structure and angle of orientation. Therefore, it is very important to find effective method for recognition and classification of digit and letter. Handwritten digit and letter recognition has various applications such as number plate recognition, extracting business card information, bank check processing, postal address processing, passport processing, signature processing etc. This paper propose a method of handwritten digit and letter recognition using OCR ,support vector machine(SVM) classifiers for classification standard sci kit learn dataset for SVM and standard template dataset for OCR is used .input image are binarized and later stray pixels are removed .the proposed method was able to obtain highest accuracy for SVM classifier.**

Keywords OCR; Skit learn;SVM;MATPLOTLIB

INTRODUCTION

Handwritten digit and letter recognition is a very important topic in the field of image processing and pattern recognition. Offline handwritten digit and letter recognition has various applications like number plate recognition, bank check processing, postal address processing, passport processing, signature processing etc. Various algorithms have been developed for classification and recognition of handwritten digits\and\letter. Recognition of handwritten digits and letter has becomes harder due to different writing styles, angle of orientation and thickness of the digit and letter. Therefore there is a need to develop algorithm that provides very high recognition accuracy and a very high speed of computation. A lot of work has been done for handwritten digit recognitionin English.Similarly huge amount of work has been done for letter recognition. For Handwritten digit and letter-recognition.
various methods are developed. In this paper we have used OCR using MATLAB and SVM using Python.

A. RELATED WORK

A lot of study has been done on recognition of handwritten digits and letter based on various feature extraction and classification methods. A number of feature extraction techniques like template matching, projection histogram and zoning are used in past. N. Ahmed et al. [1] defined Discrete Cosine Transform and proposed algorithm to compute DCT using Fast Fourier Transform. Anuj Dutt and Aashi Dutt in [2] presented a method of handwritten digit recognition using deep analysis. U Ravi Babu et al. [3] proposed handwritten digit recognition using K- Nearest Neighbour Classifier. They used structural features like water reservoir in four direction number of holes, maximum profile distance using four direction and fill hole density while achieving recognition rate of 96.94%. Muhammad Suhail Akhtar et al. [4] used wavelet packet decomposition for recognition of handwritten digits. They obtained a recognition accuracy of 97.04%. Vineet Singh and his team in [5] used Principal Component Analysis for feature reduction with single layer neural network classifier while obtaining recognition rate of 98.39%. In [6] Caiyun Ma and Hong Zhang proposed handwritten digit recognition using multi feature extraction and deep analysis and achieved recognition accuracy of 94.2%. V.N. Manjunath Aradhana et al. [7], proposed a system based on radon transform for handwritten digit recognition. Saleem Pasha and M.C. Padma in [8], used wavelet transform and structural feature for handwritten Kannada Character recognition..

B.

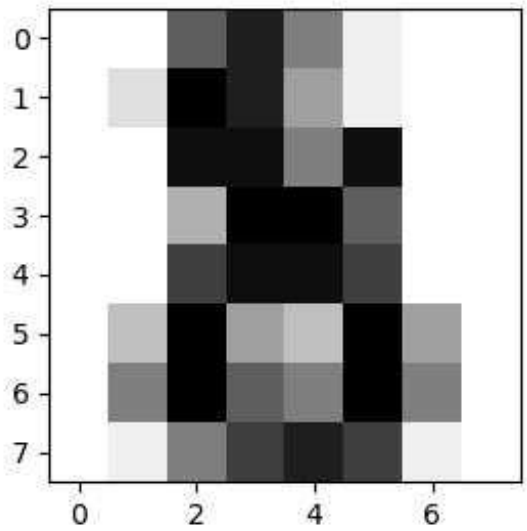
In this paper a technique of SVM classifier and OCR standard method to analyse this feature and achieve reorganisation result. The number of features are considerably reduced using this method. the accuracy of this method tested on standard template dataset and Scikit learn digits dataset. The standard template dataset contains all alphabetic and numeric templates. The Scikit learn digit dataset contains total 1779digits dataset in which training and testing dataset variable by user.

The sections of the remaining paper are follows. Section II describe dataset used for experiment. Section III explains pre-processing. Section IV explains methodology used. Section V presents classification method for digit recognition. Later sections have described acknowledgement and references used.

II. DATABASE

In OCR method we used standard template dataset and in python we used Scikit learn digit dataset.

A. IN digit datase dataset is made up of 1797 8*8 images. Each of image, like the one has shown below, is hand-written digit. In order to utilize an 8x8 figure like this, we'd have to first transform it into a feature vector with length
64



B. OCR DATASET

This dataset contains hand written words dataset collected by Rob Kassel at MIT spoken language system groups.

a clean subset of the words has been created and rasterized with normalized images of each let.

Since the first letter of each word was capitalised and rest were lower case, I removed the first letter and only used the lower-case letter the tab delimitated data file contains a line for each other with its labels, pixels values, and several additional fields listed in template file. Each letter and digit are assigned a unique integer id letters are from A-Z and digits are from 0-9.

III. PREPROCESSING

A. Binarization

For this experiment, we need both the datasets to be in binary format. For Binarization, first we compute the global threshold that can be used to convert intensity image to binary using greythresh function. In greythresh function, Otsu's method is used. In Otsu's method threshold is chosen to minimize intraclass variance of the black and white pixels. The pixel with intensity value above threshold level is represented as 1 and pixel with intensity below are represented as 0. Multidimensional arrays are converted automatically to 2d arrays using reshape function. Thus both the dataset are converted into binary format.

This preprocessing method is used in OCR (Matlab) and not in Python.

B. Python libralies

importing py plot from Matplotlib digits dataset from scikit learn.
From clf.fit function splitting the training and testing images

IV. METHODOLOGY

The objective of the project is to identify the character using set of given templates. Print the words in image in the text file. This project used the OCR and SUPPORT VECTOR MACHINE for classification of datasets and to extract output from input image as well as datasets for SVM classification.

- Each character is segmented separately.
- The character separated is scaled down to standard size [24 42] in this case.
- Each character is compared with a set of pre-defined character template set.
- Corr2 function in MATLAB is used to find co-relation factor of the segmented character with pre-defined template
- The character that gives maximum score is set to print.

For SVM and KNN

- Importing libraries Matplot lib for plots
- Import the pre-defined datasets from sklearn library.
- Get SVM from sklearn.
- Split dataset into target and digits sets.
- Get classifier as SVM put the gamma and c parameters
- Give range of data and target from last elements.
- Call for prediction from clf.fit from added data.

FOR KNN:

- import dataset in to program.
- It has 1797 records.
  - each entry is 8*8 matrix, reshape this matrix to 1*64 so as to make computation more human readable.
  - split this whole chunk of data in to training and testing data set.
  - We have split whole data in to 75% of training data and 25% of testing data
  - import classifier and train it.
  - give classifier training set, using fit function.

- KNN is lazy learner, it takes whole data while classifying each test entry.
- In other words, it doesn't learn anything at all. it just returns nearest neighbours and give us mode of all nearest neighbors..
- predict accuracy over test data.

V. CLASSIFICATION OF DIGITS

In this experiment we have used Support Vector Machine Algorithm for Classification. KNN and SVM are two widely used techniques for classification. These two classifiers were experimented on standard dataset.

Support Vector Machine Classifier (SVM)

Support Vector Machine is a supervised learning model use for classification and regression. SVM is a defined by a discriminating hyperplane. In SVM each data item is plotted as a point in n dimensional space (where n is number of features) with value of each feature being the value of a particular coordinate. A highest recognition accuracy of 97.74% is achieved with 90 features.

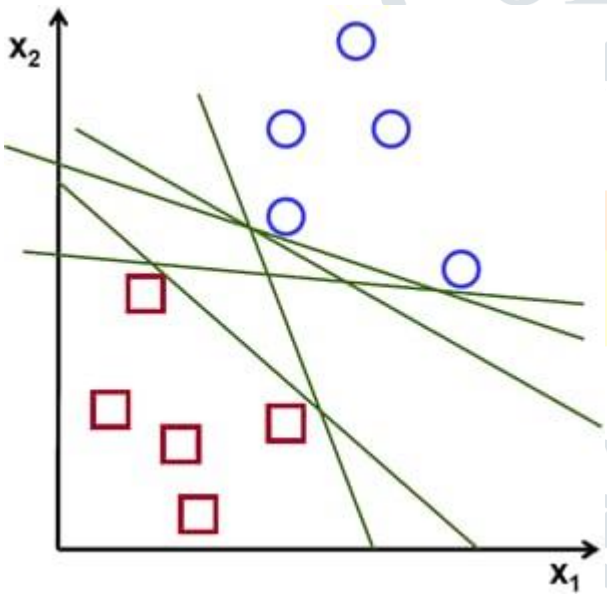


Fig.1.2OPTICAL CHARACTER RECOGNISATION

Optical Character Recognition or OCR is the process of taking an image of letters or typed text and converting it into data that computer understands. The exact mechanism that allows humans to capture an image of a paper document after which the text is extracted from that image. Hence, paper documents are easily converted into editable computer files. OCR is widely used in the field of pattern recognition and artificial intelligence. The paper describes the detailed methodology in the field of Optical Character Recognition.

FLOW OF OCR SYSTEM

In OCR processing, an input image or file is first to browse in the computer by scanning. Scanning speed will be decided by the quality of the scanner machines, paper quality, cleanness, proper setting of the OCR system. After scanning, it reads that file, decides the threshold value, analysed for light and dark areas in order to identify each alphabetic letter or numeric digit. When a character is recognized, it is converted into an ASCII code. The recognizing process is to interpret images. The library templates and configuration threshold will determine the accuracy of interpretation of the OCR.

There are three basic techniques on which OCR works: Pre-processing, Character Recognition, and Postprocessing. Before going directly to character recognition, it is needed that image we are going to used must be error free. So that accurate recognition can possible. Pre-processing implies changes if the image is not properly aligned, edges are not smooth, detects line and character, converts colour images into black-and-white images, etc. After compiling all changes to input image file, we can move to character recognition

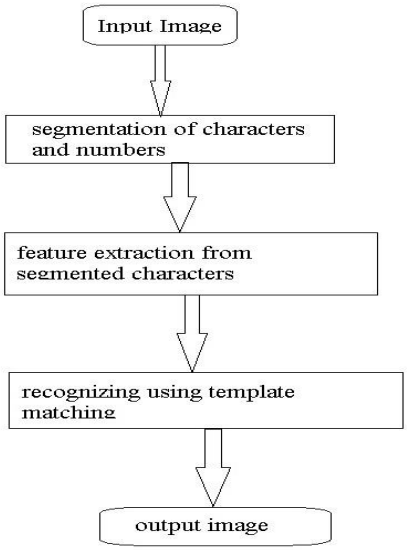


Fig.1.3 FLOWCHART OF OCR

Acknowledgment

I express my sincere gratitude to Prof.Kiran B Ingale Department of Electronics and telecommunication Engineering, VIT, for his stimulating guidance continuous encouragement and supervision throughout the project work.

References

[1] N. Ahmed, T. Natarajan and K. R. Rao, "Discrete Cosine Transform," in IEEE Transactions on Computers, vol. C-23, no. 1, pp. 90-93, Jan. 1974.

[2]Anuj Dutt, Aashi Dutt, "Handwritten Digit recognition using deep learning", International journal of Advanced Research in Computer Engineering and Technology (IJARCET), vol. 6, Issue 7, July 2017, pp. 990-997.

[3] U. R. Babu, Y. Venkateswarlu and A. K. Chintha, "Handwritten Digit Recognition Using K-Nearest Neighbor Classifier," 2014 World Congress on Computing and Communication Technologies, Trichirappalli, 2014, pp. 60-65.

[4] M. S. Akhtar and H. A. Qureshi, "Handwritten digit recognition through wavelet decomposition and wavelet

packet decomposition," Eighth International Conference on Digital Information Management (ICDIM 2013), Islamabad, 2013, pp. 143-148.

[5] V. Singh and S. P. Lal, "Digit recognition using single layer neural network with principal component analysis," Asia-Pacific World Congress on Computer Science and Engineering, Nadi, 2014, pp. 1-7.

[6] Caiyun Ma and Hong Zhang, "Effective handwritten digit recognition based on multi-feature extraction and deep analysis," 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Zhangjiajie, 2015, pp. 297-301.

[7] V. N. M. Aradhya, G. H. Kumar and S. Noushath, "Robust Unconstrained Handwritten Digit Recognition using Radon Transform," 2007 International Conference on Signal Processing, Communications and Networking, Chennai, 2007, pp. 626-629

[8] S. Pasha and M. C. Padma, "Handwritten Kannada character recognition using wavelet transform and structural features," 2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT), Mandya, 2015, pp. 346-351

[9] J. Pradeep, E. Srinivasan S. Himavathi, "Diagonal Based Feature Extraction For Handwritten Character Recognition System Using Neural Network", International Journal of Computer Science & Information Technology, vol. 3, Issue 1, pp. 27-38, Feb 2011.

[10] Anita Pal, Dayashankar Singh, "Handwritten English Character Recognition Using Neural Network", International Journal of Computer Science & Communication, vol. 1, Issue 2, pp. 141-144, July December 2010.

[11] Yann LeCun," THE MNIST database of handwritten digits" Courant Institute, NYU Corinna Cortes, Google Labs,NewYork. . hhttp://www.reserach.att.com/yann/exdb/mnist/index.html

[12] Ishani Patel, Virag Jagtap and Kailas Kale. Article: A Survey on Feature Extraction Methods for Handwritten Digits Recognition. International Journal of Computer Applications 107(12):11-17, December 2014.

[13] J. Pradeep, E. Srinivasan and S. Himavathi, "Diagonal based feature extraction for handwritten character recognition system using neural network," 2011 3rd International Conference on Electronics Computer Technology, Kanyakumari, 2011, pp. 364-368.

[14] Sudhir, Sushant & Shiva, Lambert & Wiering, Marco. (2012). Handwritten Character Classification using the Hotspot Feature Extraction Technique. 1. 261-264. 10.5220/0003712002610264

.15] O. Sarita, L. Shoemaker and M. Wiering, "A Comparison of Feature and Pixel-Based Methods for Recognizing Handwritten Bangla Digits," 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, 2013, pp. 165-169.

[16] C.-C. Chang, C.-J. Lin, "Libsvm: a library for support vector machines", ACM Transactions on Intelligent Systems and Technology (TIST), vol. 2, no. 3, pp. 27, 2011.