Handwritten Digit-Recognizer

Md. Lutfullahil Labib
Department of CSE
East West University
Dhaka, Bangladesh
2017-1-60-015@std.ewubd.edu

Tasnim Khan Nishuti
Department of CSE
East West University
Dhaka, Bangladesh
2017-1-60-041@std.ewubd.edu

Handwritten digit recognition from handwritten documents offers an interesting research problem for researchers as few research carried out on this field. The aim of this research is to investigate machine learning classification algorithm that is used to recognize different type of handwritten digits. On mainly three steps: segmentation, feature extraction and classification. In the segmentation step we used edge detection technique for segmenting dataset images using fuzzy logic. Feature extraction methods are described to take decision category of numerical digits. For feature extraction we used mRMR for feature selection, tortuosity, direction, curvatures and chain code for feature extraction and PCA for dimension reduction. In the final step, we used KNN, SVM and RFC for classification of writer attributes and recognizing handwritten digits. Classification accuracy on CSV dataset was for recognizing handwritten digits. We have used these classification algorithms to bring out the maximum accuracy rate for.

Keywords: CSV, RFC, KNN, PCA,, Chain Code, Tortuosity, Direction, Curvatures, Dimension Reduction, Edge Detection.

I. Introduction

Handwritten digit recognition is the ability of a computer to read and interpret handwritten digit from various sources such as documents, images, touch-screens and various gadgets. Postal address interpretation, bank-check processing, etc. A person's handwriting is like a fingerprint: one can copy others handwriting but can never write it in an identical way. This refers that not two individuals have exactly the same handwriting, and thus, analysis of this feature becomes more significant in various sectors like forensic biometrics, automotive industry, business field, service personnel, education, bank-check processing, signature verification, postal-address interpretation, demography-based writer identification etc.

To make machines more intelligent, the developers are diving into machine learning and deep learning techniques. A human learns to perform a task by practicing and repeating it again and again so that it memorizes how to perform the tasks. Deep learning is also very similar to this. It uses different types of neural network architectures for different types of problems.

The handwritten digit recognition is the ability of computers to recognize human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different flavors. The handwritten digit recognition is the solution to this problem which uses the image of a digit and recognizes the digit present in the image. Moreover, digit recognition also plays a significant role in authentication of documents and genuineness. Both neuroscientists and psychologists agree on the correlation between handwriting and various demographic attributes of writers.

A. Dataset

In this paper, we have tried to come up with a solution to perform tasks simultaneously and efficiently: recognize handwritings digit from various input sources/devices Artificial intelligence offers many benefits in pattern recognition and classification within the sense of emulating accommodative human intelligence to a little extent Handwriting digit Recognition is the first step to the vast field of Artificial Intelligence and Image Processing. With the advancement of technology, machines can now identify images and understand handwriting of different people which humans themselves have failed to comprehend from the image. The extracted features are used to train an artificial neural network which classifies the demographics of the author of a query. Example to accurately classify a cursive handwriting or different styles of digit recognition is tough, even for an expert. Thus, keeping all these factors and the complexity of the situation in mind we have thus decided to come up with this solution. This model shall be a new step towards digitalization and is believed to have an immense contribution in this field.

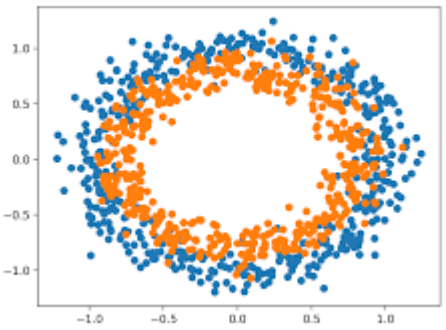

Fig. 1. Overall accuracy calculation

As for the dataset to use in the testing of the project, we have chosen the CSV Database. This database contains thousands of handwritten digits that have been used in the development of programs with a similar aim. This dataset is open for public use with no charges. It is also very convenient for our project and will help us reduce the time by using directly as a test set without having to make one ourselves

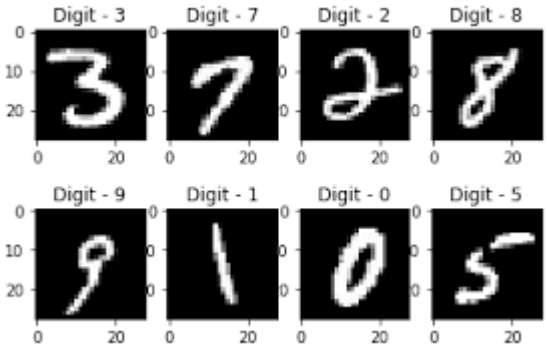

Fig. 2.handwritten digit recognition using CSV

II. Dataset Preprocessing

Pre-processing is a common name for operations with images at the lowest level of abstraction - both input and output are intensity images. The aim of pre-processing is an improvement of the image data that suppresses unwanted distortions or enhances some image features important for further processing. We have removed the duplicate points from the dataset in pre-processing step. We have used here image as data for training and testing. It will check if the training data will match or not with the digit.

III. Implementation Details

The problem of handwritten digit recognition has long been an open problem in the field of pattern classification. Several studied have shown that Neural Network has a great performance in data classification. The main objective of this paper is to provide efficient and reliable techniques for recognition of handwritten numerals by comparing various existing classification models. This paper compares the performance of five machine learning classifier models namely CSV, Random Forest, Decision Tree a Results indicate that K-NN classifier outperform Neural.

Network with significant improved computational efficiency without sacrificing performance. They both outperformed the other classifiers: Random Forest, Decision Tree and Bagging with gradient boost. We also discovered that as the training data is increasing the accuracy of the classifier is also improved. The result of this paper shows that K-NN has equally high accuracy of 93.5% compared to Neural Network of 96.8%, but K-NN achieves a processing speed with almost 10 times faster. The analysis presented in this paper suggests that the K-NN combined with preprocessing methods is capable of achieving great performance apart from Neural Network when used as a classification algorithm in offline handwritten digit recognition

A. CSV

CSV dataset is a robust and versatile classifier that is often used as a benchmark for more complex classifiers such as Artificial Neural Networks (ANN) and Support Vector Machines (SVM). Despite its simplicity, KNN can outperform more powerful classifiers and is used in a variety of application such as economic forecasting, data compression and genetics. The KNN classifier is a non-parametric and instance based learning algorithm. The advantages of using KNN are that it is robust to noisy training data and effective if the data is large.

B. Random Forest

Random forest is a supervised learning algorithm which is used for both classification as well as regression. But however, it is mainly used for classification problems. As we know that a forest is made up of trees and more trees means more robust forest. Similarly, random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result. Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes..

Step 7: Finally after calculating the accuracy we will get the obtained accuracy for our dataset. For our dataset the accuracy obtained is 93.5%.

C. Decision Tree

Decision tree characterization procedure is a standout amongst the most prevalent systems in the developing field of information mining. A method of building a decision tree from the set of samples is the method involved in the implementing decision tree algorithm. It is the form of flow chart where every non-terminal node represents the test on a particular attribute and class labels are held with the terminal node.

We have used 3 types of library here:

A. Py screenshot

The py screenshot module can be used to copy the contents of the screen to a Pillow image memory using various back-ends. Replacement for the Image Grab Module.

B. Import time

It helps to perform database backup and recover. It do the Reorganization of data or eliminate database fragmentation It moves data between Kodi databases on different platforms.

C. Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib.

preview of one image using matplotlib

```
In [15]: %matplotlib inline
         import matplotlib.pyplot as plt
         import cv2
         idx = 531
         img = X.loc[idx].values.reshape(28,28)
         print(Y[idx])
         plt.imshow(img)

         5

Out[15]: <matplotlib.image.AxesImage at 0x1ab7b9160b8>
```

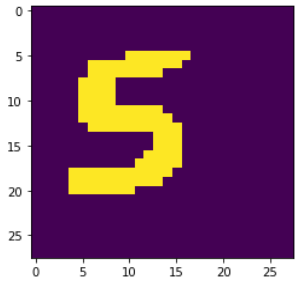

Fig 3: Preview of one image using matplotlib

IV. Results and Discussion

Among all the methods we tried on CSV dataset, a committee of three convolutional networks which are ResNet-50, VGG-5, VGG-16, (inspired and modified from kkweon's work on github), has the best performance, which has 93.5 Percentage accuracy, even better than professor Lecun's committee of 35 convolutional networks with 0.23% error rate, but not always repeatable. Tweaking the kernel sizes and numbers of convolutional layers indicates that more convolutional layers are helpful to improve the accuracy while change kernel size has little effect and the accuracy even decreased sometime. Possible reason could be that convolutional layers contribute to feature extraction and feature transform.

Fit the model using svc and also to save the mo

```
In [17]: import joblib
         from sklearn.svm import SVC
         classifier=SVC(kernel="linear", random_state=6)
         classifier.fit(train_x,train_y)
         joblib.dump(classifier, "model/digit_recognizer")

Out[17]: ['model/digit_recognizer']
```

Calculate Accuracy

```
In [18]: from sklearn import metrics
         prediction=classifier.predict(test_x)
         print("Accuracy= ",metrics.accuracy_score(prediction, test_y))

         Accuracy=  0.935
```

The CSV data set is a popular data set, on which various classification algorithms has been tested. The state of art on this data set is large Convolutional Neural Network ([1]) with unsupervised pre-training. It achieved an error rate of 0.39% on test set. However, to make a fair comparison, it's more beneficial to compare performances of different algorithms without any preprocessing on the data set. Table 1 lists the error rates of several algorithm, applied on the original data set. Note that, the performance of Nearest Neighbor Classifier will be quite good given enough training samples. In our case, it even beat neural network with 300 hidden units.

This is the final output of out handwritten digit recognition where by using the paint app we have insert the input digit and the machine return the predicted value.

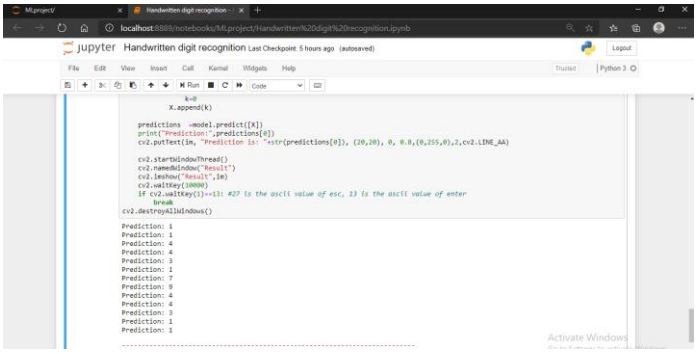

Fig 4: Predicted outputs

.

V. Conclusion and Future Work

In this paper, we have tried to come up with a solution to perform tasks simultaneously and efficiently: recognize handwritings digit from various input sources/devices Artificial intelligence offers many benefits in pattern recognition and classification within the sense of emulating accommodative human intelligence to a little extent Handwriting digit Recognition is the first step to the vast field of Artificial Intelligence and Image Processing. With the advancement of technology, machines can now identify images and recognize the numeric digits. The extracted

features are used to train an artificial neural network which classifies the demographics of the author of a query. Thus, keeping all these factors and the complexity of the situation in mind we have thus decided to come up with this solution. This model shall be a new step towards digitalization and is believed to have an immense contribution in this field.

REFERENCES

[1] M. Liwicki, A. Schlapbach, P. Loretan, and H. Bunke. Automatic Detection of Gender and Handedness from On-Line Handwriting, EURASIP Journal on Image and Video Processing, December 2014, vol. 10, pp. 1-10, 2014,

[2] A. Dutta and A. Dutta,"Handwritten digit recognition using deep learning, "International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)," vol. 6, no. 7, July 2017.

[3] S. Al. Maadeed and A. Hassaine. (2014). Automatic prediction of age, gender, and nationality in offline handwriting. EURASIP Journal on Image and Video Processing. [Online]. Available: https://doi.org/10.1186/1687-5281-2014-10.

[4] Abdul Hamid, Norhidayu&Nur Amir Sjarif, Nilam. (2017). Handwritten Recognition Using SVM, KNN and Neural Network.

[5] ÁngelMorera, Ángel Sánchez, José Francisco Vélez, and Ana Belén Moreno, "Gender and Handedness Prediction from Offline Handwriting Using Convolutional Neural Networks," Complexity, vol. 2018, Article ID 3891624, 14 pages, 2018. https://doi.org/10.1155/2018/3891624..

[6] S. Kumar and N. Shah, "False information on web and social media: A survey," arXiv preprint arXiv:1804.08559, 2018.

[7] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter, "Detection and resolution of rumours in social media: A survey," ACM Computing Surveys (CSUR), vol. 51, no. 2, p. 32, 2018.

[8] L. Wu, F. Morstatter, X. Hu, and H. Liu, "Mining misinformation in social media," Big Data in Complex and Social Networks, pp. 123–152, 2016.

[9] Z. Jin, J. Cao, Y. Zhang, J. Zhou, and Q. Tian, "Novel visual and statistical image features for microblogs news verification," IEEE Transactions on Multimedia, vol. 19, no. 3, pp. 598–608, 2017

[10] Z. Jin, J. Cao, H. Guo, Y. Zhang, and J. Luo, "Multimodal fusion with recurrent neural networks for rumor detection on microblogs," in Proceedings of the 2017 ACM on Multimedia Conference. ACM, 2017, pp. 795–816.