Handwritten Character Recognition Survey

Anantha Narayanan G.S.

Ashok Kumar N.

Bharath Kumar K.

Under the guidance of

Pradheeba U.

Abstract:

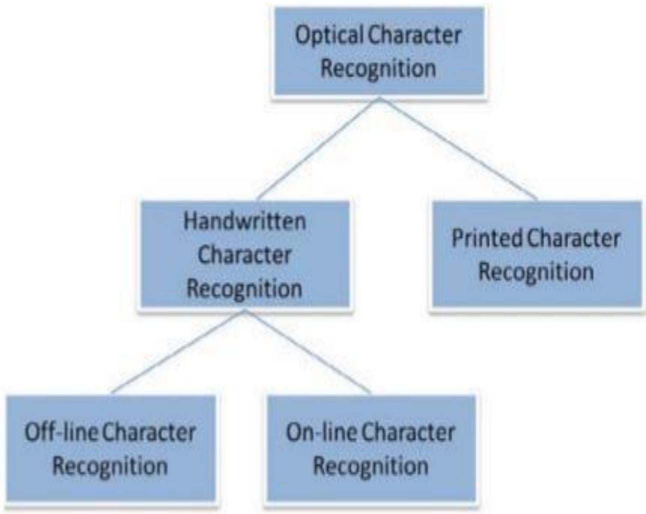
Handwritten Character Recognition is widely used to read and understand characters from images and covert them to digital content. Character recognition is a fundamental, but most challenging in the field of pattern recognition with large number of useful applications. It has been an intense field of research since the early days of computer science due to it being a natural way of interactions between computers and humans. More precisely Character recognition is the process of detecting and recognizing characters from the input image and converts it into ASCII or other equivalent machine editable form.

Introduction:

Understanding the handwritten characters or typed documents is simple to the human beings as we have the ability to learn. The same ability can be induced to the Machines also by the use of Machine Learning and Artificial Intelligence. The field which deals with this problem is called as the OCR or also known as Optical Character Recognition. It's the area of study among various fields such as recognizing of pattern, also Image vision and also AI. This is the system for changing electronic and image text into the digital character to be read by the machines.

The time used in entering the data and also the storage space required by the documents can be highly reduced by the use of OCR or in other words it can be retrieved fast. By using the OCR in banking field, legal scenarios, etc. many important and sensitive documents can be processed faster without human intervention.

OCR in advance can be inferred in two ways based on type of the text and document acquisition.



Further if we take into consideration the text type, then OCR is further of two types, HCR (Handwritten Character Recognition) which is intelligent recognition of the handwritten text and PCR (Printed Character Recognition). We need the high recognition ability due to the varying handwriting of the humans. Many a times the writing style of same individual is different at times. Further OCR is characterized into two forms as Offline and Online recognition systems based on acquiring of the documents. Offline System deals with recognizing the pre written

document acquired through various input methods. But in Online recognizing system, the writing is recognized the moment it is written. The device used for the online system is Electric pen where it is used for writing the letters or words on the device called as digitizer and on the basis of the pen movement the input is recorded.

This research provides a comprehensive comparison between different machine learning and deep learning algorithms for the purpose of handwritten character recognition while using the Support Vector Machine, Multilayer Perceptron, and Convolutional Neural Network for the same purpose The comparison between these algorithms is carried out on the basis of their accuracy, errors, and testing-training time corroborated by plots and charts that have been constructed using matplotlib for visualization.

The accuracy of any model is paramount as more accurate models make better decisions. The models with low accuracy are not suitable for real-world applications. Ex- For an automated bank cheque processing system where the system recognizes the amount and date on the check, high accuracy is very critical. If the system incorrectly recognizes a character, it can lead to major damage which is not desirable. That's why an algorithm with high accuracy is required in these real-world applications. Hence this paper deals with providing a comparison of different algorithms based on their accuracy so that the most accurate algorithm with the least chances of errors can be employed in various applications of handwritten character recognition.

This paper provides a reasonable understanding of machine learning and deep learning algorithms like SVM, CNN, and MLP for handwritten character recognition. It furthermore gives you the information about which algorithm is efficient in performing the task of character recognition. In further sections of this paper will be discussing the related work that has been done in this field followed by the methodology and implementation of all the three algorithms for the fairer understanding of them. Next, it presents the conclusion and result bolstered by various graphs. Moreover, it will also provide information about some potential future enhancements that can be done in this field. The last section of this paper contains citations and references used.

Literature Review:

Research in the region of word recognition, being done from Grimsdale in the year 1959 is soonest endeavor to perceive the handwritten character. This mid-sixty research exhibited the utilization of examination by combination strategy being proposed by the eden in 1968.He demonstrated that every single handwritten character is limited to number of schematic highlights. This hypothesis was later utilized as a part of almost all strategies for auxiliary methodologies in the region of character recognition.

K. Gaurav and Bhatia P. K [1], proposed different pre-handling systems being associated with the recognition of the characters. The procedure took a shot at the various types of pictures from a basic picture-based report to a hued and changed forces including foundation. Different systems of pre-handling and standardization like skew remedy, differentiate evacuation, commotion expulsion and numerous other upgrade procedures were recommended. They reached the decision that a solitary procedure can't be connected for preprocessing the picture. Yet additionally there were a few disparities that utilizing every one of these systems likewise can't give the best exactness comes about.

Salvador España-Boquera [2], The analysts proposed the utilization of hybrid or half plus half concealed markov show (HMM) to perceive the handwritten content in disconnected mode. The optical model's basic part was prepared with markov chain procedure and a multilayer perceptron was likewise used to gauge the probabilities.

In [3], to perceive the disconnected handwritten numerals of six prominent Indian language, a changed quadratic classifier is utilized. A similar paper likewise manages perceiving the English letters in order. For both of these, a multilayer perceptron was utilized and Boundary following and Fourier descriptors were utilized for the component extraction. By examining the shape and looking at their highlights, the characters were identified. Also, to decide the quantity of concealed layers, back spread system was utilized. With this very calculation, a recognition rate of 94% have been accounted for with less preparing time.

R. Bajaj, S. Chaudhari, L. Dey, et al [4], for grouping the Devanagari numerals, distinctive highlights like clear part, thickness and minute highlights were utilized. Additionally, to increase the recognition capacity, the paper proposes multi classifier unwavering quality for handwritten Devanagari numerals.

Sandhya Arora in [5], In this paper specifically four highlights like shadow, histogram of chain code crossing point and horizontal line fitting highlights being portrayed. Among these highlights the shadow was registered all around for picture character, the rest three were processed by partitioning the character picture into the distinctive sections. In the one useful execution utilizing the dataset of 4900 examples demonstrated the exactness rate of 90.8 % for Devanagari handwritten characters.

Nafiz Arica at al. [6] This paper gave the technique because of which it was less demanding to maintain a strategic distance from the preprocessing stage along these lines lessening the loss of imperative data. The best one proposed was calculation of capable division. What's more, the different strategies supporting this calculation were utilizing neighborhood maxima and minima, additionally other, for example, stroke tallness which turned out to be ideal and furthermore character limit. What's more, these were altogether connected on a grayscale picture. Utilizing this approach, superfluous division was decreased bit by bit.

Alongside that, the paper additionally proposed another model called shrouded markov demonstrate (HMM) preparing for estimation of worldwide and highlight space parameters alongside estimation of model parameters. Additionally, to rank the individual characters and furthermore to get the shape data, this preparation show was utilized. Additionally, by utilizing the one-dimensional portrayal of a 2-D character picture tremendously builds the energy of HMM for shape perceiving.

In [7], a technique was proposed to perceive the individually Tamil written character by utilizing the grouping in the strokes. Principally a strokes' format or shape-based portrayal is utilized spoken to as a string of shape highlights. Utilizing this strategy, the unrecognized stroke was perceived by contrasting it and a dataset of strokes by the string coordinating method in an adaptable mode. Utilizing this, an individual character was perceived by distinguishing every one of the strokes and its segments.

Analysis with problem identification:

The comparison of the algorithms (Support vector machines, Multi-layered perceptron & Convolutional neural network) is based on the characteristic chart of each algorithm on common grounds like dataset, the number of epochs, complexity of the algorithm, accuracy of each algorithm, specification of the device (Ubuntu 20.04 LTS, i5 7th gen processor) used to execute the program and runtime of the algorithm, under ideal condition.

Support Vector Machines

Support Vector Machine (SVM) is a supervised machine learning algorithm. There is generally plotting of data items in n-dimensional space where n is the number of features, a particular coordinate represents the value of a feature, we perform the classification by finding the hyperplane that distinguishes the two classes. It will choose the hyperplane that separates the classes correctly. SVM chooses the extreme vectors that help in creating the hyperplane. These extreme cases are called support vectors, and hence the algorithm is termed as Support Vector Machine. There are mainly two types of SVMs, linear and non-linear SVM.

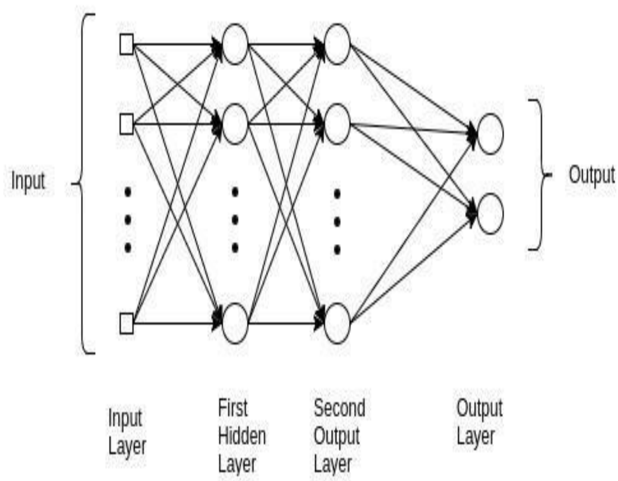
The SVM in scikit-learn [8] supports both dense (numpy.ndarray and convertible to that by NumPy.asarray) and sparse (any scipy.sparse) sample vectors as input. In sci-kit-learn, SVC, NuSVC & LinearSVC are classes capable of performing multi-class classification on a dataset. This paper has used LinearSVC for the classification of MNIST datasets that make use of a Linear kernel implemented with the help of LIBLINEAR [9].

Various sci-kit-learn libraries like NumPy, matplotlib, pandas, Sklearn & seaborn have been used for the implementation purpose. Firstly, download the MNIST datasets, followed by loading it and reading those CSV files using pandas.

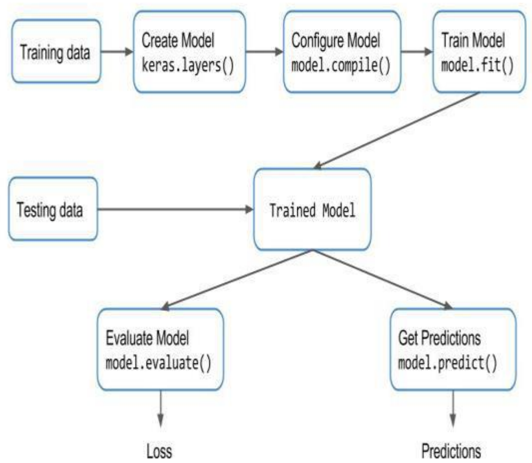
After this, plotting of some samples as well as converting into matrix followed by normalization and scaling of features have been done. Finally, create a linear SVM model and confusion matrix that is used to measure the accuracy of the model [10].

Multilayer Perceptron

A multilayer perceptron (MLP) is a class of feedforward artificial neural networks (ANN). It consists of three layers: input layer, hidden layer & output layer. Each layer consists of several nodes that are also formally referred to as neurons and each node is interconnected to every other node of the next layer. In basic MLP there are 3 layers but the number of hidden layers can increase to any number as per the problem with no restriction on the number of nodes. The number of nodes in the input & output layer depends on the number of attributes & apparent classes in the dataset respectively. The particular number of hidden layers or numbers of nodes in the hidden layer is difficult to determine due to the model erratic nature and therefore selected experimentally. Every hidden layer of the model can have different activation functions for processing. For learning purposes, it uses a supervised learning technique called backpropagation. In the MLP, the connection of the nodes consists of a weight that gets adjusted to synchronize with each connection in the training process of the model. [11]



The implementation of Handwritten character recognition by Multilayer perceptron [12] which is also known as feedforward artificial neural network is done with the help of Keras module to create an MLP model of Sequential class and add respective hidden layers with different activation function to take an image of 28x28 pixel size as input. After creating a sequential model, the Dense layer of different specifications and Drop out layers as shown in the image below is added. The block diagram is given here for reference. Once training and test data are available, one can follow these steps to train a neural network in Keras. The paper uses a neural network with 4 hidden layers and an output layer with 10 units (i.e. total number of labels). The number of units in the hidden layers is kept to be 512. The input to the network is the 784-dimensional array converted from the 28×28 image. A sequential model is used for building the network. In the Sequential model, anyone can just stack up layers by adding the desired layer one by one. This paper used the Dense layer, also called a fully connected layer since it is building a feedforward network in which all the neurons from one layer are connected to the neurons in the previous layer. Apart from the Dense layer, the ReLU activation function is added, which is required to introduce non-linearity to the model. This will help the network learn non-linear decision boundaries. The last layer is a softmax layer as it is a multiclass classification problem [13].



Convolutional Neural Network

CNN is a deep learning algorithm that is widely used for image recognition and classification. It is a class of deep neural networks that require minimum pre-processing. It inputs the image in the form of small chunks rather than inputting a single pixel at a time, so the network can detect uncertain patterns (edges) in the image more efficiently. CNN contains 3 layers namely, an input layer, an output layer, and multiple hidden layers which include Convolutional layers, Pooling layers(Max and Average pooling), Fully connected layers (FC), and normalization layers [14]. CNN uses a filter (kernel) which is an array of weights to extract features from the input image. CNN employs different activation functions at each layer to add some non-linearity [15]. Further, we observe the height and width decrease while the number of channels increases. Finally, the generated column matrix is used to predict the output [16].

The implementation of handwritten digit recognition by Convolutional Neural Network [17] is done using Keras. It is an open-source neural network library that is used to design and implement deep learning models. From Keras, Sequential class is used which allows anyone to create a model layer-by-layer. The dimension of the input image is set to 28(Height), 28(Width), 1(Number of channels). Next, the model is created whose first layer is a Conv layer [18]. This layer uses a matrix to convolve around the input data across its height and width and extract features from it. This matrix is called a Filter or Kernel. The values in the filter matrix are weights. 32 filters each of the dimensions (3,3) with a stride of 1 is used. Stride determines the number of pixels shifts. Convolution of filter over the input data gives activation maps whose dimension is given by the formula: $((N + 2P - F)/S) + 1$ where N= dimension of input image, P= padding, F= filter dimension and S=stride. In this layer, Depth (number of channels) of the output image is equal to the number of filters used. To increase the non-linearity, an activation function that is ReLU [19] is used. Next, another convolutional layer is used in which 64 filters of the same dimensions (3,3) with a stride of 1 and the Relu function is applied.

Next, to these layers, the pooling layer [20] is used which reduces the dimensionality of the image and computation in the network. Furthermore, MAX-pooling is employed which keeps only the maximum value from a pool. The depth of the network remains unchanged in this layer. the pool-size (2,2) is kept with a stride of 2, so every 4 pixels will become a single pixel. To avoid overfitting in the model, the Dropout layer [21] is used which drops some neurons which are chosen randomly so that the model can be simplified. The probability of a node getting dropped out is set to 0.25 or 25%. Following it, Flatten Layer [22] is used which involves flattening i.e. generating a column matrix (vector) from the 2-dimensional matrix. This column vector will be fed into the fully connected layer [23]. This layer consists of 128 neurons with a dropout probability of 0.5 or 50%. After applying the ReLu activation function, the output is fed into the last layer of the model that is the output layer. This layer has 62 neurons that represent classes (numbers from 0 to 9 and alphabets from a to z and A to Z) and the SoftMax function [24] is employed to perform the classification. This function returns probability distribution over all the 62 classes. The class with the maximum probability is the output.

Conclusion:

After implementing all the three algorithms that are SVM, MLP & CNN. This paper compares their accuracies and execution time with the help of experimental graphs for perspicuous understanding. It has taken into account the Training and Testing Accuracy of all the models stated above. After executing all the models, it has found that SVM has the highest accuracy on training data while on testing dataset CNN accomplishes the utmost accuracy. Additionally, it has compared the execution time to gain more insight into the working of the algorithms. Generally, the running time of an algorithm depends on the number of operations it has performed. So, deep learning models have been trained up to 30 epochs and SVM models according to norms to get the apt outcome. SVM took the minimum time for execution while CNN accounts for the maximum running time. The SVM gave an efficient training accuracy with a testing accuracy of above 90 percent. But to just make sure that the model is flawless, this paper shuffles the dataset by using a train test split and it still gives the accuracy above 90 percent making the model perfect. Therefore this paper includes one of the accuracies given by the SVM.

FUTURE ENHANCEMENTS

The future development of the applications based on algorithms of deep and machine learning is practically boundless. In the future, work on a denser or hybrid algorithm than the current set of algorithms with more manifold data to achieve the solutions to many problems can be done.

In future, the application of these algorithms lies from the public to high-level authorities, as from the differentiation of the algorithms above and with future development one can attain high-level functioning applications which can be used in the classified or government agencies as well as for the common people, these algorithms can be used in hospitals application for detailed medical diagnosis, treatment and monitoring the patients, it can be used in a surveillance system to keep tracks of the suspicious activity under the system, in fingerprint & retinal scanners, database filtering applications, Equipment checking for national forces and many more problems of both major and minor category. The advancement in this field can help to create an environment of safety, awareness & comfort by using these algorithms in day to day application and high-level application (i.e. Corporate level or Government level). Application-based on artificial intelligence & deep learning is the future of the technological world because of their absolute accuracy and advantages over many major problems.

References:

1.K. Gaurav and Bhatia P. K., “Analytical Review of Preprocessing Techniques for Offline Handwritten Character Recognition”, 2nd International Conference on Emerging Trends in Engineering & Management, ICETEM, 2013.

2. Salvador España-Boquera, Maria J. C. B., Jorge G. M. and Francisco Z. M., “Improving Offline Handwritten Text Recognition with Hybrid HMM/ANN Models”, IEEE Transactions on Pattern Analysis

3. U. Pal, T. Wakabayashi and F. Kimura, “Handwritten numeral recognition of six popular scripts,” Ninth International conference on Document Analysis and Recognition ICDAR 07, Vol.2, pp.749-753, 2007.

4. Reena Bajaj, Lipika Dey, and S. Chaudhury, “Devnagari numeral recognition by combining decision of multiple connectionist classifiers”, Sadhana, Vol.27, part. 1, pp.-59-72, 2002.

5. Sandhya Arora, “Combining Multiple Feature Extraction Techniques for Handwritten Devnagari Character Recognition”, IEEE Region 10 Colloquium and the Third ICIIS, Kharagpur, INDIA, December 2008.

6. Nafiz Arica, and Fatos T. Yarman-Vural, ―Optical Character Recognition for Cursive Handwriting,‖ IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.24, no.6, pp. 801-113, June 2002.

7. K. H. Aparna,, Vidhya Subramanian, M. Kasirajan, G. Vijay Prakash, V. S. Chakravarthy, Sriganesh Madhvanath, "Online Handwriting Recognition for Tamil", IWFHR, 2004, Proceedings. Ninth International Workshop on Frontiers in Handwriting Recognition, Proceedings. Ninth International Workshop on Frontiers in Handwriting Recognition

8. “Support vector machines”, https://scikit-learn.org/stable/modules/svm.html

9. “LIBSVM”, https://en.wikipedia.org/wiki/LIBSVM

10. rishikakushwah16, “SVM_digit_recognition_using_MNIST_dataset”, GitHub, [Online]. Available: https://github.com/rishikakushwah16/SVM_digit_recognition_using_MNIST_dataset

11. “Crash Course On Multi-Layer Perceptron Neural Networks” https://machinelearningmastery.com/neural-networks-crash-course/

12. Flame-Atlas, “Handwritten_Digit_Recognition_using_MLP”, GitHub, [Online]. Available:

13. https://github.com/Flame-Atlas/Handwritten_Digit_Recognition_using_MLP/blob/master/ANN.py

14. “An Introduction to Convolutional Neural Networks” Available: https://www.researchgate.net/publication/285164623_An_Introduction_to_Convolutional_Neural_Networks

15. “Activation Functions: Comparison of Trends in Practice and Research for Deep Learning”, Chigozie Enyinna Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall. Available: https://arxiv.org/pdf/1811.03378.pdf

16. “Basic Overview of Convolutional Neural Network” https://medium.com/dataseries/basic-overview-of-convolutional-neural-network-cnn-4fcc7dbb4f17

17. dixitritik17, “Handwritten-Digit-Recognition”, GitHub, [Online]. Available: https://github.com/dixitritik17/Handwritten-Digit-Recognition

18. “Image Classification using Feedforward Neural Network in Keras ” https://www.learnopencv.com/image-

classification-using-feedforward-neural-network-in-keras/

19. “How Do Convolutional Layers Work in Deep Learning Neural Networks” available: https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/

20. “Rectified Linear Unit (ReLU)”, https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

21. “Pooling Layers for Convolutional Neural Networks”, https://machinelearningmastery.com/pooling-layers-for-convolutional-neural-networks/

22. “Dropout in (Deep) Machine learning”, https://medium.com/@amarbudhiraja/https-medium-com-amarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machine-learning-74334da4bfc5

23. “Basic Overview of Convolutional Neural Network (CNN)”, https://medium.com/dataseries/basic-overview-of-convolutional-neural-network-cnn-4fcc7dbb4f17