

Handwritten Digit Recognition System Using Neural Network.

Vinay Kumar
Email id:-vinay17225@gmail.com
School of CSA
REVA University Bengaluru,
Karnataka, India
PIN: 560064

Adithya V
Assistant Professor
Email: adithya.v@reva.edu.in
School of CSA
REVA University Bengaluru,
Karnataka, India
PIN: 560064

ABSTRACT: - Digit Recognition is a noteworthy and important issue. As the manually written digits are not of a similar size, thickness, position and direction, in this manner, various difficulties must be considered to determine the issue of handwritten digit recognition. The uniqueness and assortment in the composition styles of various individuals additionally influence the example and presence of the digits. It is the strategy for perceiving and arranging transcribed digits. It has a wide range of applications, for example, programmed bank checks, postal locations and tax documents and so on.

INTRODUCTION: -

Recognition is identifying or distinguishing a thing or an individual from the past experiences or learning. Similarly, Digit Recognition is nothing but recognizing or identifying the digits in any document. Digit recognition framework is simply the working of a machine to prepare itself or interpret the digits. Handwritten Digit Recognition is the capacity of a computer to interpret the manually written digits from

various sources like messages, bank cheques, papers, pictures, and so forth and in various situations for web based handwriting recognition on PC tablets, identifying number plates of vehicles, handling bank cheques, digits entered in any forms etc. Machine Learning provides various methods through which human efforts can be reduced in recognizing the manually written digits. Deep Learning is a machine learning method that trains computers to do what easily falls into place for people: learning through examples. With the utilization of deep learning methods, human attempts can be diminished in perceiving, learning,

recognizing and in a lot more regions. Using deep learning, the computer learns to carry out classification works from pictures or contents from any document. Deep Learning models can accomplish state-of-art accuracy, beyond the human level performance. The digit recognition model uses large datasets in order to recognize digits from distinctive sources. Handwriting recognition of characters has been around since the 1980s. The task of handwritten digit recognition, using a classifier, has extraordinary significance and use such as – online digit recognition on PC tablets, recognize zip codes on mail, processing bank check amounts, numeric sections in structures filled up by hand (for example - tax forms) and so on. There are diverse challenges faced while attempting to solve this problem. The handwritten digits are not always of the same size, thickness, or orientation and position relative to the margins. The main objective was to actualize a pattern characterization method to perceive the handwritten digits provided in the MINIST data set of images of handwritten digits (0-9)

OBJECTIVE:-

This work is conducted by using Machine learning concepts. Before going deep into the topic, we must know about some of these concepts. A human can easily solve and recognize any problem, but this is not the same in the case of a machine. Many techniques or methods should be implemented to work as a human. Apart from all the advancements that have been made in this area, there is still a significant research gap that needs to be filled. Consider, for example, online handwriting recognition vs offline recognition. In online

handwriting recognition of letters, an on-time compilation of letters is performed while writing because stroke information is captured dynamically. Whereas, in offline recognition, the letters aren't captured dynamically. Online handwriting recognition is more accurate when compared to offline handwriting recognition because of the lack of information. Therefore, there can be research done in this area to improve offline handwriting recognition.

LITERATURE SURVEY

Anuj Dutta in his paper demonstrated that utilizing Deep Learning systems, he had the capacity to get an extremely high measure of accuracy. By utilizing the convolutional Neural Network with Keras and Theano as backend, he was getting a accuracy of 98.72%. In addition, execution of CNN utilizing Tensorflow gives a stunningly better consequence of 99.70%. Despite the fact that the complication of the procedure and codes appears to be more when contrasted with typical Machine Learning algorithms yet the accuracy he got is increasingly obvious. In a paper published by Saeed [1] AL-Mansoori Multilayer Perceptron (MLP) Neural Network was implemented to recognize and predict handwritten digits from 0 to 9. The proposed neural system was trained and tested on a dataset achieved from MNIST. An early notable attempt in the area of character recognition research is by Grimsdale in 1959. The origin of a great deal of research work in the early sixties was based on an approach known as analysis-by-synthesis method suggested by Eden in 1968. The great importance of Eden's work was that he formally proved that all handwritten characters are formed by a finite number of schematic features, a point that was implicitly included in previous works. This notion was later used in all methods in syntactic (structural) approaches of character recognition. [2], Velappa Ganapathy, and Kok Leong Liew they proposed a method in which first multi-scale neural training with modifications in the input training vectors is adopted to acquire its advantage in training higher resolution character images and then selective thresholding using minimum distance technique is proposed to increase the level of accuracy of character recognition. A simulator program (a GUI) is designed in such a way that the characters can be located on any spot on the blank paper in which the characters are written. The results show that such methods with moderate level of training epochs can produce accuracies of at least

85% and more for handwritten upper case English characters and numerals [3] Mathias M. Adankon, Mohamed Cheriet the LS-SVM classifier, like other kernel machines, gives a poor generalization when the hyper parameters are not tuned efficiently. The authors proposed a model selection strategy for the LS-SVM which is a variant of the popular SVM classifier. They formed model selection using the empirical error criterion through the LOO procedure. They applied an algorithm on a handwriting recognition problem, which gave promising results. Compared with the SVM, the sparse LS-SVM classifier, empowered by model selection based on the empirical error criterion and the LOO procedure, achieved higher performance. They conclude from this that the sparse LS-SVM with model selection would be an interesting alternative classifier for the SVM in pattern recognition systems [4]

Advantages of Proposed system

1. Easily identifies trends and patterns -

- Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

- With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus software; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Continuous Improvement

- As ML algorithms gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data you have keeps

growing, your algorithms learn to make more accurate predictions faster.

4. Handling multi-dimensional and multi-variety data

- Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

5. Wide Applications

- You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.
- Modules Used in Research paper

TensorFlow

1. TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.
2. TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

NumPy

1. NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.
2. It is the fundamental package for scientific computing with Python. It contains various features including these important ones:
 - A powerful N-dimensional array object
 - Sophisticated (broadcasting) functions
 - Tools for integrating C/C++ and Fortran code

- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using NumPy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

Pandas

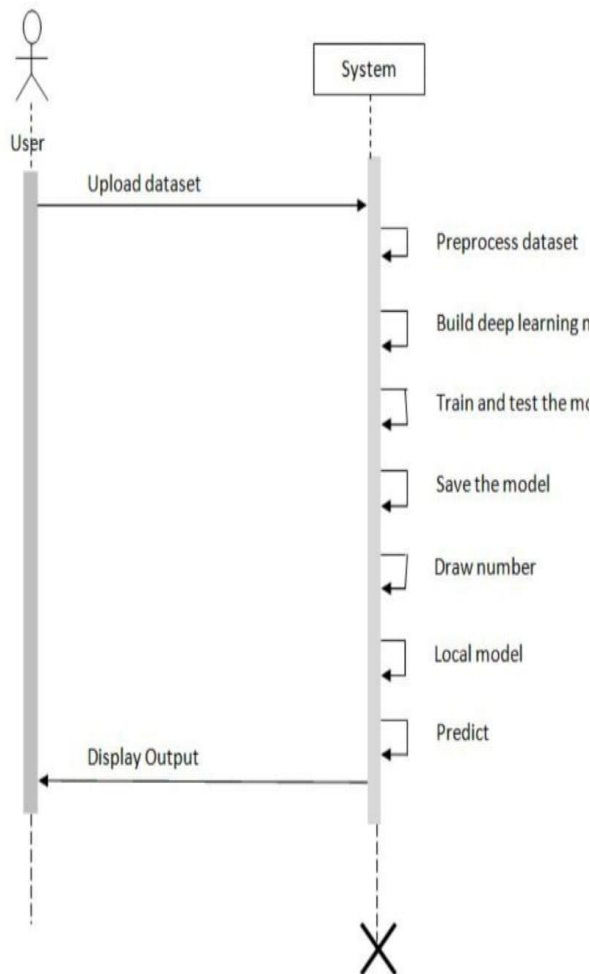
- Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem.
- Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Matplotlib

- Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.
- For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a

set of functions familiar to MATLAB users.

Architecture Diagram



IMPLEMENTATION

- First, we are going to import all the modules that we are going to need for training our model. The Keras library already contains some datasets and MNIST is one of them. So we can easily import the dataset and start working with it. The `mnist.load_data()` method returns us the training data, its labels and also the testing data and its labels.
- The image data cannot be fed directly into the model so we need to perform some

operations and process the data to make it ready for our neural network. The dimension of the training data is (60000,28,28). The CNN(convolutional Neural Network) model will require one more dimension so we reshape the matrix to shape (60000,28,28,1).

- Now we will create our CNN model in Python data science project. A CNN model generally consists of convolutional and pooling layers. It works better for data that are represented as grid structures; this is the reason why CNN works well for image classification problems. The dropout layer is used to deactivate some of the neurons and while training, it reduces over fitting of the model. We will then compile the model with the Adadelta optimizer.
- The `model.fit()` function of Keras will start the training of the model. It takes the training data, validation data, epochs, and batch size.
- We have 10,000 images in our dataset which will be used to evaluate how good our model works. The testing data was not involved in the training of the data therefore, it is new data for our model. The MNIST dataset is well balanced so we can get around 99% accuracy.
- Now for the GUI, we have created a new file in which we build an interactive window to draw digits on canvas and with a button, we can recognize the digit. The Tkinter library comes in the Python standard library. We have created a function `predict digit()` that takes the image as input and then uses the trained model to predict the digit.

- Then we create the App class which is responsible for building the GUI for our app. We create a canvas where we can draw by capturing the mouse event and with a button, we trigger the `predict digit()` function and display the results.

Results



Manual Testing

Manual testing includes testing a software manually, i.e., without using any automated tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behaviour or bug. There are different stages for manual testing such as unit testing, integration testing, system testing, and user acceptance testing. Testers use test plans, test cases, or test scenarios to test a software to ensure the completeness of testing. Manual testing also includes exploratory testing, as testers explore the software to identify errors in it.

Unit Testing

This type of testing is performed by developers before the setup is handed over to the testing team to formally execute the test cases. Unit testing is performed by the respective developers on the individual units of source code assigned areas. The developers use test data that is different from the test data of the quality assurance team.

FUTURE ENHANCEMENT

The complex recognition problem associated with handwriting is an interesting topic for future research areas. For instance, when some anonymous pieces of handwritten character are found at a crime site, and it is possible to automatically identify that

the writer may be a “left-handed man,” that would reduce the set of suspects to be investigated.

BIBLIOGRAPHY

- [1] A. Dutta, Handwritten digit recognition using deep learning, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), vol. 6, no. 7, July 2017.
- [2] Al Mansoori, Somaya, and Abdelaali Hassaine, Automatic prediction of age, gender, and nationality in offline handwriting. EURASIP Journal on Image and Video Processing, no. 1 2014.
- [3] Velappa Ganapathy, and Kok Leong Liew, “Handwritten Character Recognition Using Multi scale Neural Network Training Technique”, Proceedings of World academy of Science, Engineering and Technology, vol. 29, ISSN 1307-6884, May (2007)
- [4] Mathias M.Adankon, Mohamed Cheriet, “Model selection for the LS-SVM. Application to handwriting recognition”, Pattern Recognition, vol.42, pp.3264-3270, (2009)
- [5] 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.
- [6] 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 and Machine Intelligence, Vol. 33, No. 4, April 2011.
- [7] 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.
- [8] Anita Pal & Dayashankar Singh, “Handwritten English Character Recognition Using Neural,” Network International Journal of Computer Science & Communication. Vol. 1, No. 2, July-December 2010, pp. 141-144.
- [9] J. Pradeep, E. Srinivasan and S. Himavathi, “Diagonal Based Feature Extraction For Handwritten Alphabets Recognition System Using

Neural Network”, International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 1, Feb 2011.

[10] A. Brakensiek, J. Rottland, A. Kosmala and J. Rigoll, “Offline Handwriting Recognition using various Hybrid Modeling Techniques & Character N-Grams”, Available at <http://irs.ub.rug.nl/dbi/4357a84695495>.

[11] Reena Bajaj, Lipika Dey, and S. Chaudhury, “Devnagari numeral recognition by combining decision of 35 multiple connectionist classifiers”, Sadhana, Vol.27, part. 1, pp.-59-72, 2002.

[12] 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.

[13] M. Hanmandlu, O.V. Ramana Murthy, “Fuzzy model based recognition of handwritten numerals”, pattern recognition, vol.40, pp.1840-1854, 2007.

[21] The MNIST database of handwritten digits. •
Homepage: <http://yann.lecun.com/exdb/mnist/> •
Source code: `tfds.image_classification.MNIST`.