**IDEATION ON HANDWRITTEN DIGIT RECOGNITION SYSTEM**

**INTRODUCTION:**

Handwritten Digit Recognition (HDR) is the process of converting images of handwritten digit into digital format. A lot of money is wasted on converting the information that is in paper to digital format. This problem can be solved by using HDR. The heart of our project lies within the ability to develop an efficient algorithm that can recognize the handwritten digits which are scanned and sent as input by the user. The goal of this paper is to observe the variation of different algorithms that can classify the handwritten digits using different hidden layers, various number of epochs and to make a comparison based on the accuracy. This experiment is performed using the Modified National Institute of Standards and Technology (MNIST) dataset.

**PROBLEM DEFINITION**

The total world is working with the various problems of the machine learning.The goal of the machine learning is to factorize and to manipulate the real life data and the real life part of the human interaction or complex ideas or the problems in the real life.The most curious of those is Handwritten Character Recognition because it is the building block of the human certified and the classification interaction between other humans.

So, the goal was to create an appropriate algorithm that can give the output of the handwritten character by taking just a picture of that character. If one asks about Image processing then this problem can’t be solved because there can be a lot of noises in that taken image which can’t be controlled by human.The main thing is when human write a handwritten character or for our case digit he has no single idea whether he has to draw it in the circulated pixels or just same as a standard image given .A machine can do that but not the human.So by matching only the pixels one can’t recognize that.

The idea of machine learning lies on supervised data.Machine learning algorithm fully dependent on modeled data .If someone models the Image directly, the model will get a lot of flatten values because that picture can be drawn with various RGB format or with various pixels which can’t be modeled accurately due to noise.

So, for this project one has to create a model by image processing and the machine learning. Both the techniques will be needed because these two techniques will enhance the technique of the machine learning and that can shape this project

**NEED OF THE SOFTWARE**

The total project lies with a great computation speed and by a online server where run and compilation done quickly. All the packages were imported that were needed for the software online. We need the tools to be imported also.

**Tensorflow:**

TensorFlow is an end-to-end open source package in python for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

**Python 3.7*:***

Python is widely used worldwide and is an advanced programming language. It was introduced primarily to stand out from the code, and its language structure enables application developers to express ideas in a few lines of code. Python is a programming language that gives you the opportunity to work faster and link frames more effectively.

**Anaconda3 5.3.1 :**

Ruser also shows the accuracy level predicted by the model. Uploading images with different adjustments and more.  Anaconda is a free and open source platform.  Python and R system for acquiring logical knowledge such as information science, AI applications, large-scale data processing, scientific research, and more. Anaconda comes with over 1,400 packages just like the Conda package and the visual environment director, called Anaconda Navigator, so it takes a while to find a way to launch the library freely. Anaconda Navigator is a graphical UI (GUI) embedded in the Anaconda distribution that enables clients to send requests and manage conda packages, situations and channels without using command line direction.

**Matplotlib:**

Matplotlib is used to plot model accuracy and loss in a graphical view in this project. Matplotlib is a Python 2D plotting library that produces pleasant figures for the publication throughout platforms in a variety of hardcopy formats and interactive environments. Matplotlib can be used in Python scripts, Python and IPython shells, Jupyter notebook, Web software servers, and four interface toolkits for graphical users.

**DATASET:**

Modified National Institute of Standards and Technology (MNIST) is a database which is freely available for handwritten digits and is standard for machine learning algorithms. It is similar to TIDigit which is a database of speech created by Texas Instruments, which tasks in speech recognition [9]. For our project, MNIST dataset is used. In this dataset, the images of digits were taken from a variety of scanned documents in which each image is Greyscaled and of 28\*28 pixels. It uses 60,000 images to train the network and 10,000 images to evaluate hoe accurately the network learned to classify the images. Some of the sample images of the MNIST dataset are shown below.

**ARCHITECTURE:**

1. **Pre-Processing**:

Pre-processing is a part of HDR. If there are some rules like a box for each digit then, it will be much easier to detect the boundaries. The fundamental motivation behind pre-processing is to take off noise filtering, smoothing, and standardization. Binarization converts a Greyscaled image into a binary image.

1. **Feature Extraction:**

Different type of algorithms used for feature extraction have different types of error rate. The errors made by each separate algorithm does not overlap, so combining all these methods lead to a perfect recognition rate and also helps to reject the ambiguous digits recognition and improve the recognition rate of misclassified digits that can be recognized by humans [10].

1. **Classification and Recognition**:

In the classification and recognition step, the extracted feature vectors are given as single input values to each classifier. CNN Convolution layer and the subsampling layer can have various different layers [17]. The down sampling layer is also known as pooling layer [19]. The image is divided into small segments of small areas, and a value is calculated for each area. Then the calculated values are rearranged in sequence to form a new image [7]. This process is similar to fuzzy filter, which can increase the robustness of image feature withdrawal [20]. Extracted features are combined and are defined using the following four classifiers:

**D.Training and Testing:**

Using the fit() method, a model can be trained. In order to see the skill of the trained model, test data is used as a validation dataset. Finally, to evaluate a model, the test dataset is used. Training is less complex because each module is designed to handle a specific subproblem.

**V. METHODOLOGY**

* Gathering, examining, and cleaning the data (data exploration)
* Choose a model and measure the evaluation (Neural network)
* Training
* Evaluate the models on a metric and compare them to the benchmark
* Compare different Machine Learning algorithms based on their accuracy of predicting the digit

**VI. IMPLEMENTATION**

The data which is already collected can be used for extracting the features of each digit. The availability of more powerful machine learning algorithms introduces an efficient and better approach to solve this problem. The project is divided two modules as shown below

1. **Pre-processing module**

* **Read the image**

OpenCV is a Machine Learning library that is used to read and manipulate images. The image is read and then stored in multiple copies for performing different operations. After reading the image is plotted in its shape to make sure it is read perfectly

* **Converting an RGB image to a Greyscale image**

An RGB image that is three-dimensional is converted to a Greyscale image that is one-dimensional. A BGR image is a three dimensional image (w, h, c)

* **Remove noise**

Gaussian blur is applied to the greyscale image to remove noise in the image

* **Object Detection**

The standard step for object detection is Otsu thresholding

* **Finding and drawing contours**

The findContours() and drawContours methods() are used for finding and drawing boundaries of the detected object in an image.

1. **Training & Testing module:**

Any model learns by using past data and Machine Learning algorithms. It learns from the past data by feature extractions. For training the model, a sequence of hidden layers is created with some nodes in each layer. Then, we compile the model by ‘categorical\_crossentropy’as loss function, ‘adam’ as an optimizer , and ‘accuracy’ as metrics. Then, the model is trained using CNN and tested by giving new handwritten digit images that are not present in the dataset.

**This project entitles some different new ideas**

1. Image Processing

2. Machine learning

3. Activation Functions

4. Statistical predictive modeling

5. Optimiser into the programming

6. Text analysis

7. Digit extraction Features.

**CONCLUSION:**

Convolutional Neural Network gets trained from the real-time data and makes the model very simple by reducing the number of variables and gives relevant accuracy. In our project, we used CNN with some libraries like Keras, Matplotlib, CV2, Tensorflow to get the maximum accuracy. A study within the paper reveals that there’s still scope of enhancing the algorithms similarly as enhancing the speed of recognition of characters.