**Detailed Description**

1. **System Architecture**

Preprocessing

Import Dataset

Initialize a Modal

Build a Multi layer CNN

Train Modal

Evaluate Modal

Classify Digit

1. **UML/ER Diagrams**

Yet to do

1. **Module Description**
   1. **METHODOLOGY**

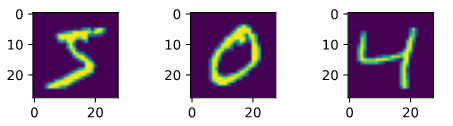
**3.1.1 Dataset**

Handwritten character recognition is a research area that already contains detailed ways of implementation which include major learning datasets. MNIST dataset (Modified National Institute of Standards and Technology database) is the subset of the NIST dataset. MNIST is a combination of two of NIST's databases Special Database 1 and Special Database 3, which consist of digits written by high school students and employees of the United States Census Bureau, respectively. MNIST has a total of 70,000 handwritten digit images, out of which 60,000 are used for training set and 10,000 are used for test set. Each image has a dimension of 28x28 pixel and are anti-aliased. All these images have corresponding label values which tells the value of the digit.

**3.1.2 Convolutional Neural Network**

Convolutional Neural Network is a subset of Deep learning, which is commonly used for Image recoginition and classification. It is a class of deep neural networks that require minimum pre-processing. It takes the image as an input in the form of small chunks rather than taking one pixel at a time, so the network can detect irregular patterns in the image more efficiently. CNN contains 3 layers namely, an input layer, an output layer, and multiple hidden layers. Hidden layers include three types of layers

1. **Convolutional layer** - This layer uses a filter (also called as kernel) which is an array of weights to extract features from the input image. One layer can have many filters.
2. **Pooling layer** - This layer reduces the dimensions of the data coming from Convolutional layer which in return reduces the computations, number of parameters, reduces overfitting and therefore making the entire process much faster. Typically, the pooling layer is inserted between convolutional layers. It discards the activations of previous layers and hence forcing the next convolutional layers to learn from a limited variety of data
3. **Fully connected layer**  - Output of previous layers is flattend and sent into this layer, which then uses that too classify the image into label.
   1. **IMPLEMENTATION**
      1. **Data Visualization**



Sample images from MNIST Dataset

**3.2.2 Data Preprocessing**

Initially, the dateset is seperated into four different categories namely trainX, trainY, testX, testY such that X represents the feature and Y represents the label. Since a digit always comes within a category of 0 to 9, the label can be converted from a simple digit to a array of size 10 where the index which represents the digit is given 1 and rest of the indexs are given a value of 0. For example 5 would be represented as [0,0,0,0,0,1,0,0,0,0]. Then a normalization process is done ,where each pixel is converted from a value range of 0 to 255 to value range of 0 to 1. Pixels are converted into float type and then divided by 255.

1. **DFD**

**Level 1**

User

CNN for Digit Recognition

Output

Image

Processing

**Level 2**

Yet to do