1. INTRODUCTION

1.1 PROJECT PROFILE

Title	: Malayalam Handwritten Character Recognition using Convolutional Neural Network
Type	: Deep learning Research and Development Project
Objective	: To detect Malayalam handwritten characters
Project Co-ordinator	: Prof. Baby Syla Asst. Professor, Department of Computer Applications, College of Engineering, Trivandrum
Project Guide	: Prof. Jose T Joseph Asst. Professor & Head, Department of Computer Applications, College of Engineering, Trivandrum

1.2 PROJECT OVERVIEW

Optical character recognition is the mechanical or electronic conversion of images of typed, handwritten or printed text into machine encoded format. Optical character recognition has been successfully implemented in many areas which greatly reduce manual work of encoding physical document to machine encoded format. Different methods are used in OCR for different languages. Recognizing handwritten text is harder than recognizing printed texts as different handwritten scripts may having different style of writings ,slants etc.

Malayalam characters are complex due to their curved nature and there are characters which are formed by the combination of two characters. These along with the presence of ‘chillu’ make recognizing Malayalam characters a challenging task. This challenge could almost meet by implementing a CNN model

Convolutional neural networks (CNN) is a popular deep learning method and is a state of art for the image recognition. this project provides a command line interface for processing a real time Malayalam handwritten document, CNN has achieved a breakthrough in the IMAGENET challenge 2011. The CNN used in the challenge was Alexnet and gave an error rate of 16% in comparison to 25% in 2010. From then on it was CNN all the way. It is very suitable for finding patters in data, As of now this project is developed to detect 44 Malayalam characters. for complete functioning of Malayalam handwritten character recognition application some more advanced researches have to done be done in this project.

1.2.1 Aims of the Project

This project aims to develop a system to recognize handwritten Malayalam characters using Convolutional Neural Network (CNN) which is a very popular deep learning technique. Malayalam is one among the scheduled languages in India and is the official language in kerala. Malayalam characters are having a curved nature and there are many glyphs which has very similar looks and some characters are the combination of other characters. All these difficulties makes Malayalam characters are

hard to detect as it requires deep machine learning models to classify every characters of it.

1.3 DOMAIN INFORMATION

This project is developed using Keras , a high level API for deep learning, Tensorflow , an open source deep learning library, and OpenCV which is open source computer vision library, all these libraries are available in their python bindings. Python programming language is used for development. Jupyter Notebook is used for better interactive visualization and block wise running of python codes during development.

1.3.1. TensorFlow

TensorFlow is an open source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms, and from desktops to clusters of servers to mobile and edge devices. Originally developed by researchers and engineers from the Google Brain team within Google's AI organization, It comes with strong support for machine learning and deep learning and the flexible numerical computation core is used across many other scientific domains. It provides a variety of different toolkits that allow you to construct models at your preferred level of abstraction. We can use lower-level APIs to build models by defining a series of mathematical operations. Alternatively, w can use higher-level APIs to specify predefined architectures, such as linear regressors or neural networks.

TensorFlow consists of the following two components a graph protocol buffer and runtime that executes the distributed graph. These two components are analogous to Python code and the Python interpreter. Just as the Python interpreter is implemented on multiple hardware platforms to run Python code, TensorFlow can run the graph on multiple hardware platforms, including CPU, GPU, and (Tensor Processing Unit).TPU is an accelerated integrated circuit developed by google specifically for processing Tensors .for this project Tensorflow is used as the backend for creating CNN model

1.3.2 Keras

Keras is a high-level neural networks API, written in Python and capable of running on top of Tensorflow, CNTK or Theano. It was developed with a focus on enabling fast experimentation. It was developed to make implementing deep learning models as fast and easy as possible for research and development. It runs on Python 2.7 or 3.5 and can seamlessly execute on GPUs and CPUs given the underlying frameworks. It is released under the permissive MIT license. In this prokect Keras is used as the abstraction layer running on top of TensorFlow. The CNN model creation can be done in Keras with a few lines of code and is readable also

1.3.3 OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos etc.this project uses OpenCV for doing tasks such as dataset creation,dataset augmentation and pre processing and real time testing of images containing handwritten Malayalam text.

1.3.4 Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library. All these aforementioned libraries are available in their python libraries which this project is made use of.

2. PROBLEM DEFINITION AND METHODOLOGY

2.1 PROBLEM DEFINITION

Optical character recognition has leveraged its capabilities to reduce tedious manual work of converting images containing characters to texts for recent decades. OCR is the mechanical or electronic conversion of images of typed, handwritten or printed text into machine encoded format. Optical character recognition has been successfully implemented in many areas which greatly reduce manual work of encoding physical document to machine encoded format. Different methods are used in OCR for different languages. Recognizing handwritten text is harder than recognizing printed texts as different handwritten scripts may having different style of writings ,slants etc. this project is an attempt to create a Handwritten character recognition system for Malayalam language

Malayalam is one among the twenty two scheduled languages in India and is the official language in the state of Kerala, where more than 95% people use Malayalam for communication. Malayalam characters are complex due to their curved nature and there are characters which are formed by the combination of two characters. These along with the presence of ‘chillu’ make recognizing Malayalam characters a challenging task.

This project aims to develop a system to recognize handwritten Malayalam characters using Convolutional Neural Network (CNN) which is a very popular deep learning technique. Malayalam is one among the scheduled languages in India and is the official language in Kerala. Malayalam characters are having a curved nature and there are many glyphs which have very similar looks and some characters are the combination of other characters. All these difficulties makes Malayalam characters are hard to detect as it requires deep machine learning models to classify every characters of it. It needs better deep learning models to correctly classify the characters in the language. an interface should be developed which enables the user to input an image of the handwritten Malayalam script and do proper preprocessing segmentation and give the Unicode Malayalam characters as output

2.2 METHODOLOGY

2.2.1 Deep learning

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called Artificial neural networks (ANN). In deep learnig we don't need to do handcrafted feature extraction like we do in classifiers like SVM and Random Forest. The ANN extracts features by its own. For character classification This project uses Convolutional neural network (CNN) which is a very polpular ANN for finding patterns in data.

2.2.1.1 *Convolutional Neural Networks (CNN)*

Convolution neural network algorithm is a special kind of feed forwarderd multilayer perceptron which is basically an Artificial Neural Network (ANN). It has proven its design for identification of patterns from two dimensional data. It has successfully been applying in image classification, natural language processing etc.The main difference between a CNN and an ANN is that CNN uses parameter sharing which makes the computation a lot more easier. A typical CNN Always has one input layer, convolution layers, pooling layers ,reLU(Rectified Linear Unit) layers,fully connected layers and one output layer. In addition to this, the number of layer in a CNN is subjected to change according to the classification requirements . a simple cnn is shown in figure 2.1.

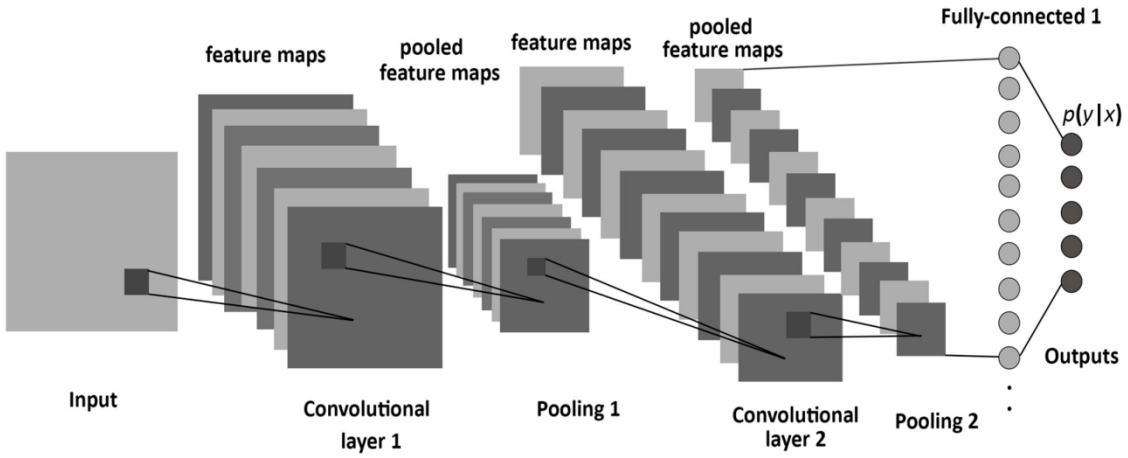

Figure 2.1 Convolutional Neural Network

CNNs use variation of multilayer perceptrons designed to require minimal preprocessing. They are also known as shift invariant or space invariant artificial neural networks. Convolutional layers apply a convolution operation to the input, passing the result to the next layer. Convolutional networks may include local or global pooling layers, which combine the outputs of neuron clusters at one layer into a single neuron in the next layer. This is done for down sampling a feature map obtained as the output of a convolutional layer. The final fully connected layers connects every neuron in the previous layer to the next layer. So the final downsampled featuremap is stretched into a single vector and is given as input to the fully connected layer. The output of the final fully connected layer contains the required number of classes.

3. REQUIREMENT ANALYSIS AND SPECIFICATION

3.1 REQUIREMENT DEFINITION

Requirement definition is the high abstract description of requirements. Requirements may be functional or non-functional.

3.1.1 Functional Requirements

The functional requirements identified in this project are

1. The system should be able to detect Malayalam handwritten characters, to do the character classification task a deep learning model should be trained with dataset of Malayalam characters.
2. The system should enable the user to input an image containing Malayalam handwritten characters. For that a command line interface should be created.
3. The system should be able to provide the user an output consists of Malayalam Unicode characters.

3.1.2 Non Functional Requirements

3.1.2.1 *Performance Requirements*

The main performance requirements that the product should satisfy are:

1. Accuracy: Accuracy in prediction of handwritten characters of different styles and slants
2. Proper training: The training set used for training should be a wise choice of images. It should be able to cover all the possible shape of a particular character.
3. Algorithms : The algorithms used for different stages should be efficient for proper result

3.1.2.2 *Quality Requirements*

The most important quality requirements that the system should satisfy are:

1. Scalability: The software will meet all of the functional requirements without an unexpected behavior.

2. Maintainability: The system should be maintainable. It should keep backups to atone for system failures, and should log its activities periodically.
3. Reliability: The acceptable threshold for down-time should be long as possible. i.e. mean time between failures should be large as possible. And if the system is broken, time required to get the system back up again should be minimum.
4. Testability: The proposed system should be properly tested under various light conditions and quality conditions of the input image to ensure that system performs well in all the circumstances

3.2. SYSTEM SPECIFICATION

3.2.1. Hardware Requirement

The minimum hardware requirement of computer is

1. Processor : Intel i7 Processor
2. Storage : 10 GB Hard Disk space
3. Memory : 8 GB RAM
4. GPU : Nvidia Quadro K2200

3.2.2. Software Requirement

The minimum software requirement of computer is

1. Operating system : Linux
2. Language : Python
3. Frameworks : Keras, TensorFlow,OpenCV
4. Editors : Jupyter Notebook,Vim

4. SYSTEM ANALYSIS

4.1 EXISTING SYSTEM

Malayalam Script digitalization is a common requirement in areas of literature publications, for film screenwriters, news reporters, writers etc. it has a number of other uses also. In the existing system the handwritten documents has to be explicitly converted into digital form by reading it with human eye and type it manually to a computer. This is a time consuming and a tedious task which require lot of manual effort to complete the intended exercise. With the support of a smart character recognition mechanism, almost all hindrances can be eradicated

4.2 PROPOSED SYSTEM

The product can be effectively used in situations where we need to digitalize a Malayalam handwritten script , by using an intelligent system to recognize the handwritten characters and make a digital equivalent of the scripts we can greatly reduce the manual effort of character recognition and data entry for digitalizing a handwritten document .Malayalam Script digitalization are a common requirement in areas of literature publications, for film screenwriters , news reporters etc. With a support of a smart character segmentation mechanism, the system can be used to meet the requirements of all these areas

4.2.1 Advantage of Proposed System

1. We can avoid the hassle of manually converting a Malayalam handwritten text
2. It greatly reduces the time of converting a Malayalam handwritten texts to Unicode characters
3. It is portable. we can port this system into an android application with less modifications on current implementation

5. DESIGN

5.1 SYSTEM ARCHITECTURE

The following System Architecture of the proposed system gives an overall idea about the project.

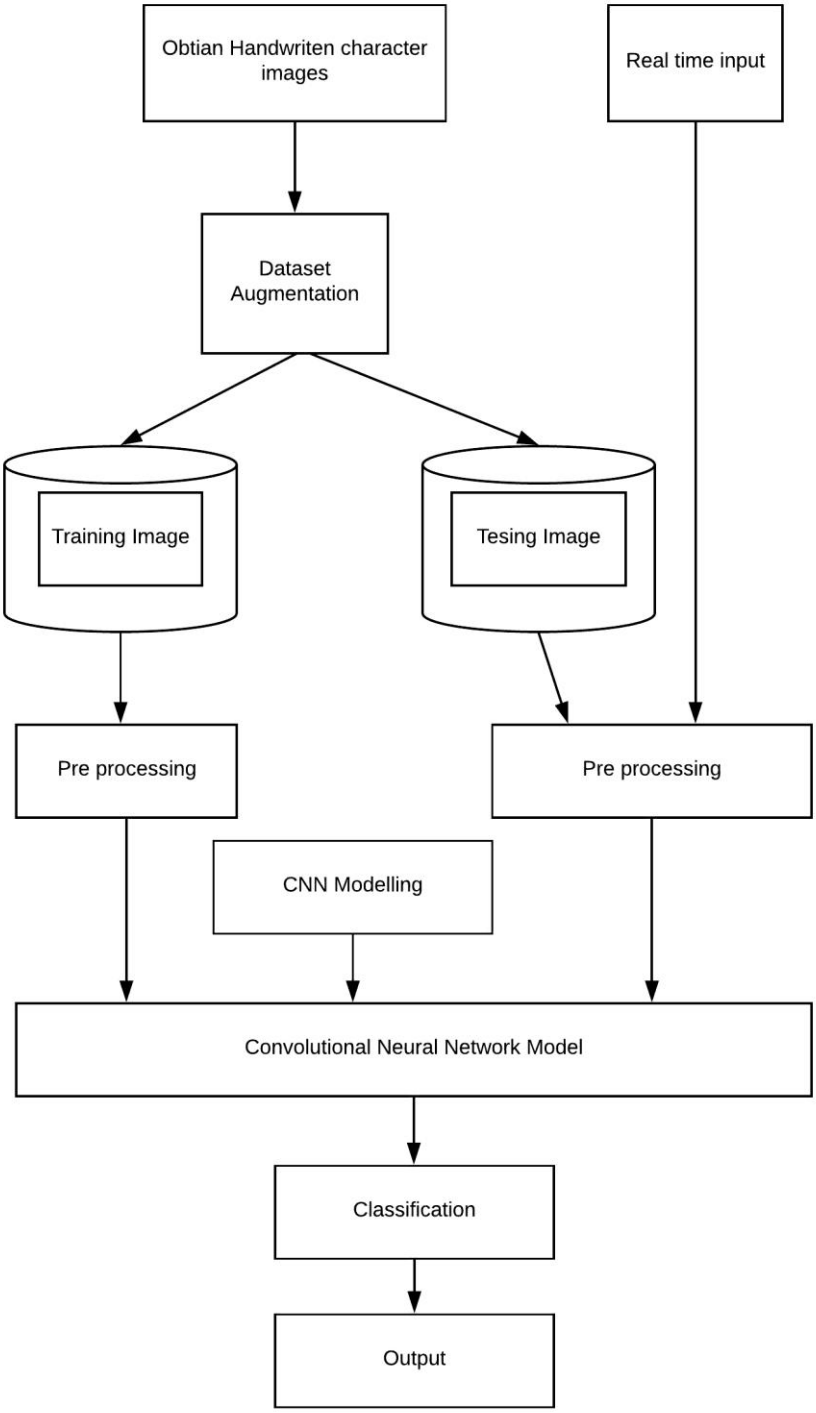

Figure 5.1 System Architecture

Initially a dataset is created by acquiring images of Malayalam handwritten characters. This is the most time consuming task in this project, here a dataset of 44 Malayalam characters is created, the CNN model needs a large number of images to train in order to get better level of accuracy. For this, dataset augmentation is performed. Dataset augmentation does the task off replicating images by making some changes in the original image. Affine transformations are applied in each input image such as Translation, Rotation, scaling and shearing. Thus the model can learn a wide variety of possible representation of a character image. Then the images are pre processed to remove the noise present in the image, the dataset are divided into training and testing sets , the training set along with corresponding labels is then given to the CNN architecture we just created, after successful completion of the training process, the model is tested using the test dataset which is already labeled.

During model deployment, the image containing Malayalam handwritten characters is inputted and after doing pre processing it is given to our model and predictions are made , this predictions are then mapped to corresponding Malayalam characters.

5.2 MODULES DESCRIPTION

The system mainly consists of four phases. They are:

1. Dataset creation and augmentation
2. Creating a CNN model
3. Training the CNN model
4. Deploy the model

5.2.1 Dataset Creation and Augmentation

Creating a dataset is time consuming and requires a lot of effort. There is no open source dataset available for handwritten Malayalam characters. For this project the dataset was collected from a private organization and modified. The dataset consisted images of 44 Malayalam characters. A large dataset is required for training the CNN. In order to attain this, the images that are already obtained is modified and transformed to get a large number of variations. Figure 5.1.2 shows the overall flow of this process.

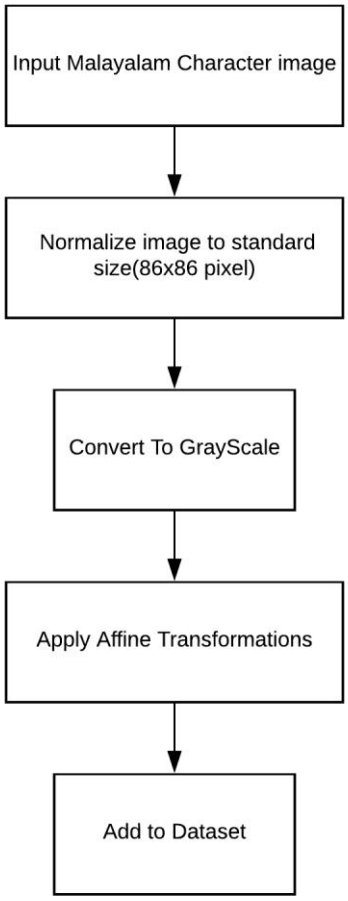

Figure 5.2 Dataset creation

Affine transformation is a linear mapping method that preserves points, straight lines, and planes. Sets of parallel lines remain parallel after an affine transformation. Different translations are used to augment the dataset Gaussian smoothing is the result of blurring an image by a Gaussian function. The visual effect of this blurring technique is a smooth blur resembling that of viewing the image through a translucent screen. Salt-and-pepper noise is a form of noise sometimes seen on images. It presents itself as sparsely occurring white and black pixels. Contrast and brightness level of an image is changed. After data augmentation a dataset of 91,902 images were obtained.

5.2.2 Creating a CNN Model

Convolutional neural network layer types mainly include three types, namely Convolutional layer, pooling layer and fully-connected layer apart from the input and output layer. There are many standard CNN architectures available which are proven

their classification capabilities in many tasks ,eg. LeNet , Alexnet,VGGNet,Inception etc. all these architecture differ due to the difference in number of hyper parameters of the network such as number of convolution layers, pooling layers ,fully connected layers ,the number of filters used in each layers, dropout rate at each layer, L2 or L1 regularization parameters, activation function type (ReLU, Sigmoid, Tanh etc.). After some researches and tries, a CNN architecture is defined for our classification task.

5.2.1.1 defining the CNN Architecture

The input size of images dataset is fixed to 86x86x1, so the initially the network is designed to occupy tensors(3D arrays here) of shape [86,86,1].The architecture of CNN model is shown in Figure 5.1.3.This model uses 3x3 filters for convolution,2x2 filters for max-pooling and ReLu(Rectified Linear Unit) as activation function in Non linearity layer. Batch normalization is applied to get better results, Adam is used as optimizer for gradient descend algorithm to perform, During training .a single image of a Malayalam character with one-hot encoded label passes through all the layers and according to the applied gradient descent strategy here, the weights are updated. The output layer contains 44 classes each represents a character in our dataset

5.2.3 Training the CNN Model

The model has to be trained with the dataset created for this we need to prepare the dataset to input and labels format so that the model understands it. for this. The dataset is processed and a list is created which consists of numpy array of images along with the one-hot encoded label. The processed dataset is divided into training set and Testing set in the ratio of 80:20 . and again the training set is further divided into training and validation sets in the same 80:20 ratio, there were 58816 images for training,14705 images for validation and 18381 images for testing. Validation set used here to test during the training time itself so that we can see whether our model overfitts or not during each epoch. An epoch is the complete iteration of training over the entire dataset. To reduce the complexities of memory inefficiencies while training the entire dataset is divided into batches and each batch is given to the network for training.

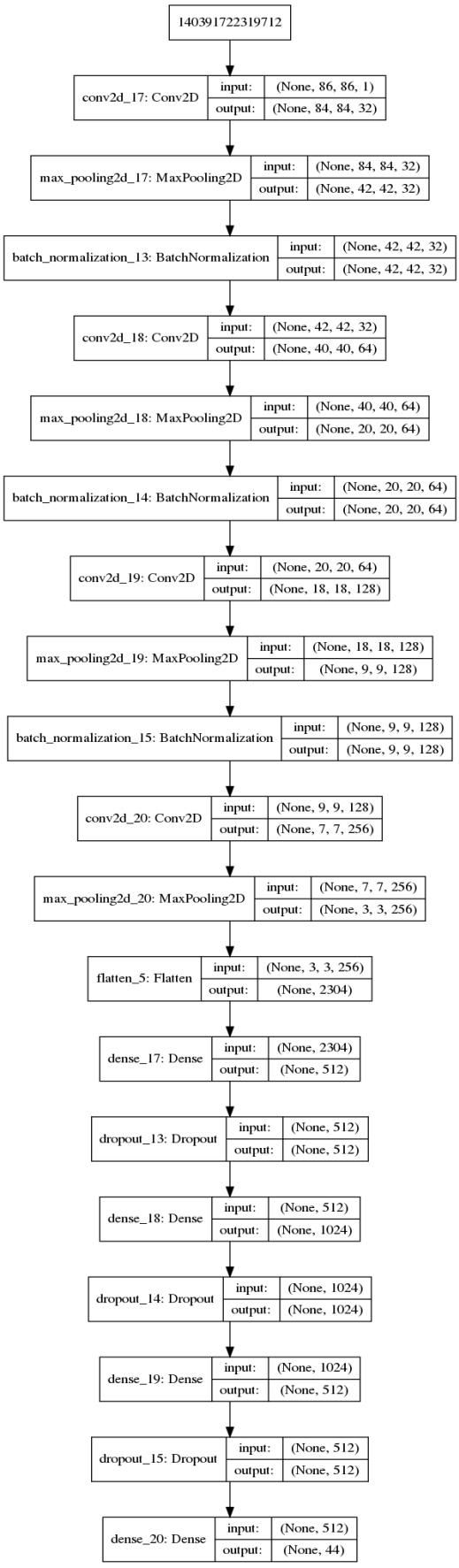

Figure 5.3 CNN Model

5.2.4 Deploy The Model

After successful training, the Model has to be deployed so that a user can make use of it the flow chart in figure 5.1.4 describes how the model can be implemented in real time. The image of Malayalam handwritten character has to be read .after applying proper segmentation mechanisms words are separated and then characters are separated. Each character that is segmented are given to model and prediction is made. For each predicted class the character is mapped to corresponding Malayalam Unicode character.

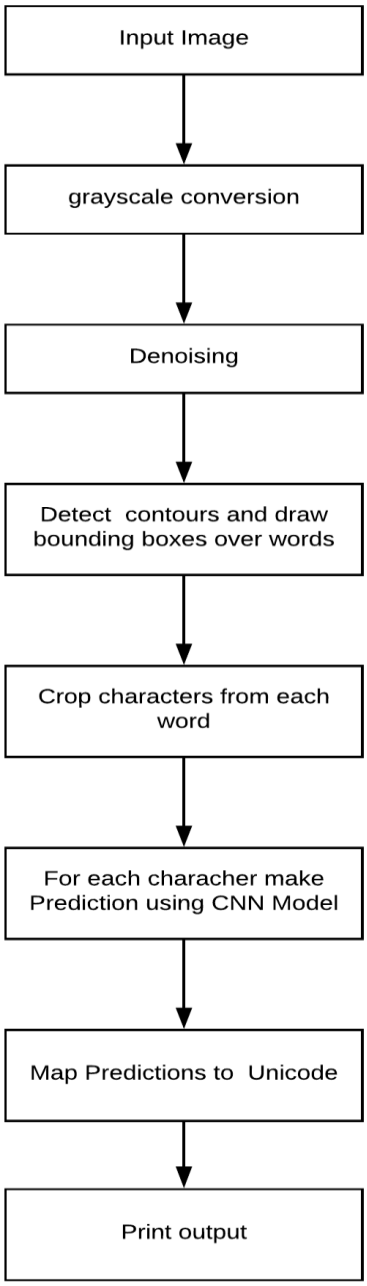

Figure 5.4 Model Deployment

5.3 DATA FLOW DIAGRAMS

5.3.1 Level 0 DFD

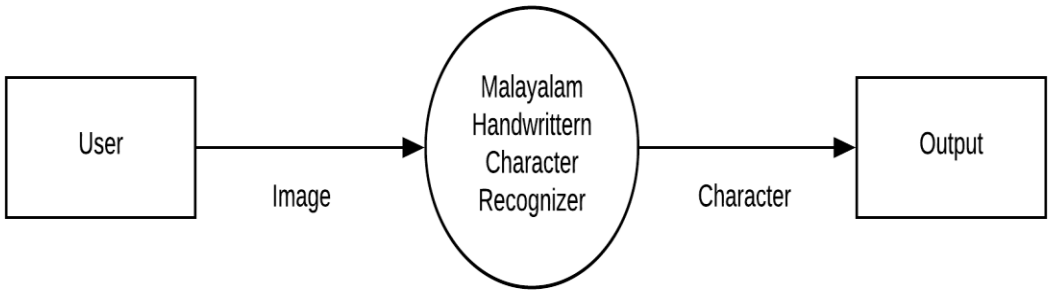

Fig 5.5 Level 0 DFD

5.3.2 Level 1 DFD

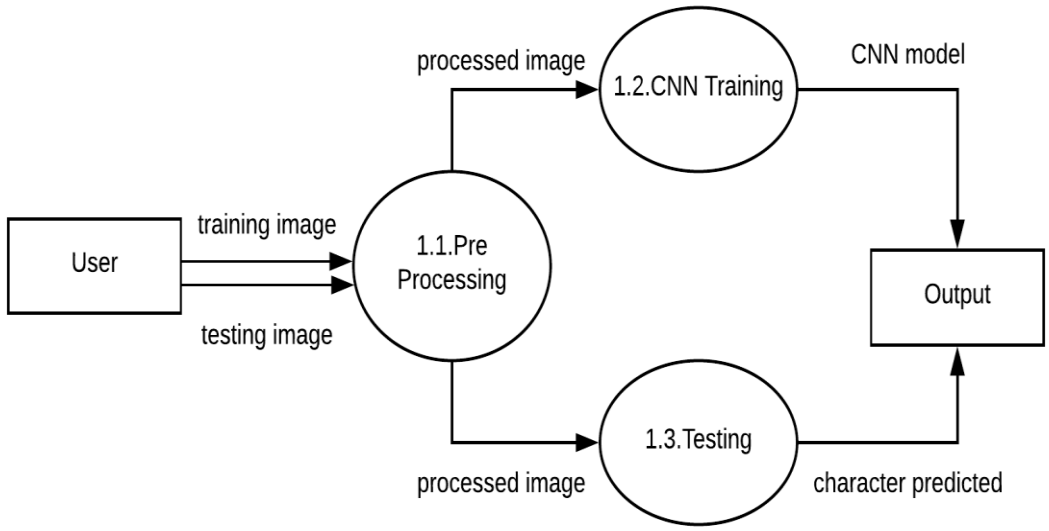

Fig 5.6 Level 1 DFD

6. SYSTEM IMPLEMENTATION

6.1 SYSTEM IMPLEMENTATION

6.1.1 Preparing Dataset for CNN

The dataset created should be prepared to the form which the CNN model understands. For that the whole 91902 images where processed to create a single list of numpy arrays along with the labels. The labels are given to the model as one-hot encoded arrays .here the array will have size 44 and each label is identified by the index which has element 1.

The dataset contains 44 directories and each contains images of a single Malayalam character .The name of each directory is the class of the corresponding character inside. The algorithm for this is shown here

1. Iterate through directory 1 to directory 44
2. For each directory take the directory name as the label name and convert the label number to corresponding one-hot array
3. Iterate through each directory and append the image and label to a list

6.1.2 Segmentation Algorithm

The segmentation algorithm is used to process the image of a Malayalam handwritten document. Each words are segmented from the document and a list of words are created and each word is further segmented into characters this algorithm also does the sorting of each words in the way a user want .The algorithm is as follows.

1. Read image
2. Convert image into grayscale
3. Apply morphologyEx operation with structuring element of kernel sizes suitable to get the words as a single contour
4. Store the contour bouding boxes as a list

5. For each bounging box ,apply morphologyEx operation with structuring element of small kernel size
6. Find contours and draw bounding boxes
7. Sort the bounding boxes in the increasing order of x axis
8. Crop each bounding box coordinate from input image and store it in a list

6.1.3 Deployment of CNN Model

This algorithm takes the list of list containing numpy array of images characters of each word from segmentation algorithm .The algorithm is as follows:

1. Input the image list
2. Load the CNN model
3. Iterate through each sub list in list
4. For each array in sub list call model.predict() method
5. Map the predicted class to corresponding character
6. Print predictions for each sub list

7. TESTING

All possible testing methods done for the project

7.1 UNIT TESTING

Test cases and results

Sl. No	Input/Procedures	Expected Results	Actual Results	Pass/Fail
1	Image Pre processing	Pre processing is done	Same as expected	pass
2	De-noising the data	Noise removed data	Same as expected	pass
3	Segment The characters	Character segmentation is done	Same as Expected	pass

Table 7.1 unit testing

7.2 INTEGRATION TESTING

Test cases and Results

Sl. No	Input/Procedures	Expected Results	Actual Results	Pass/Fail
1	Preparing data and label from dataset	Data with labels	Same as expected	pass
2	Mapping the prediction to character	Malayalam Unicode character is returned	Same as expected	pass

Table 7.2 Integration testing

7.3 SYSTEM TESTING

Test cases and results

Sl. No	Input/Procedures	Expected Results	Actual Results	Pass/Fail
1	Image of Malayalam handwritten character	Malayalam Unicode character is returned	Fair	pass

Table 7.3 System testing

7.4 TESTING THE CNN MODEL

The Dataset we have is divided as training set and testing set. We have 18,381 images specifically for testing the CNN model that we have created. The rest of images from 90,902 are used for training. The test images are already labeled and the CNN model is tested with this data. The testing accuracy found was 97.16, ie. Out of 18,381 images 17,858 images were correctly classified as their true labels.

8. RESULTS AND DISCUSSIONS

This project uses Convolutional neural network to classify Malayalam handwritten characters. For that a CNN model is created and the hyper parameters are tuned to get the increased accuracy. The model gave a training accuracy of 97.14 with loss 0.3718, validation accuracy of 96.04 with loss 0.3242 and a testing Accuracy of 97.29. A confusion matrix is created which shows the true labels vs predicted label proportion. figure 9.1 shows the confusion matrix

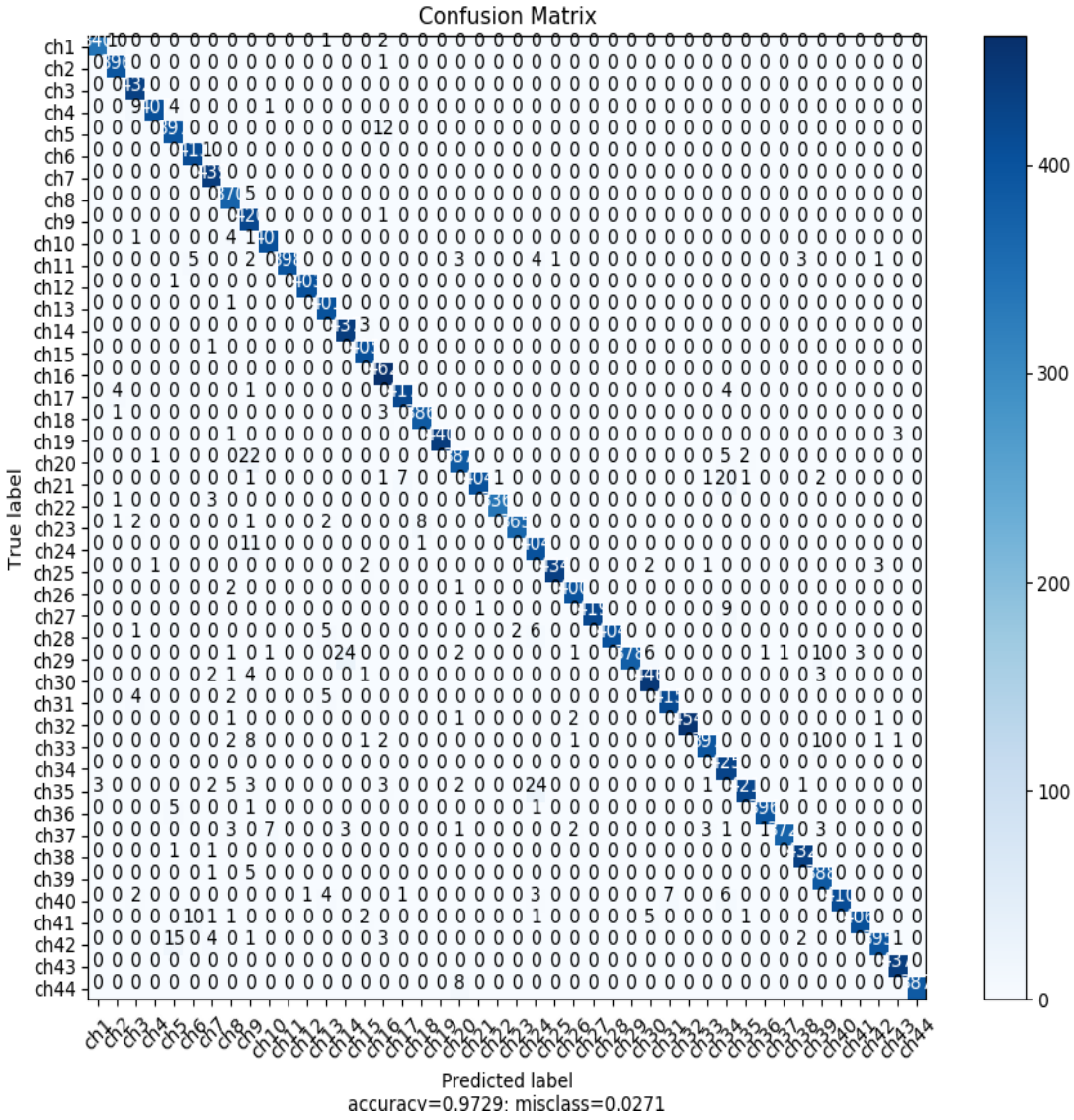

Fig 8.1 Confusion matrix

9. CONCLUSION & FUTURE ENHANCEMENTS

9.1 CONCLUSION

Handwritten character recognition is a difficult task as the characters usually has various appearances according to different writer, writing style and noise. Researchers have been trying to increase the accuracy rate by designing better features, using different classifiers and combination of different classifiers. These attempts however are limited when compared to CNN. CNNs can give better accuracy rates but it has some problems that need to be addressed. Malayalam characters are complex due to their curved nature. Here this project built a system that recognizes Malayalam handwritten characters using Convolutional neural network approach. This project will greatly help to reduce the digitalization work of any Malayalam handwritten document

OCR has a wide variety of real time applications. It can be used for office automation. This work implements a handwritten Malayalam character recognition system. The characters. Both Sample generation and CNN modeling are time consuming tasks and the later also requires a CUDA enabled GPU for parallel processing. Preprocessing helps to remove the undesired qualities of an image and hence can play an important role in increasing the role. So is the sample generation process that reduces overfitting. The drop out layer also reduces overfitting while also decreasing the overall training time. CNN has proved to be the state-of-the-art technique for other languages and hence provides the chance for giving higher accuracy rate for Malayalam characters too.

9.2 FUTURE ENHANCEMENTS

This model can classify 44 Malayalam characters .There are more symbols in Malayalam language .it has to be improved to classify all the characters and symbols that is present in Malayalam language. Malayalam language has a peculiarity that unlike English language a character may have different meaning with respect to its position.In current system , if the input image containing words that are connected the current system may not give intended results. Some more researches have to be done on this. The state based model such as RNN (Recurrent neural networks) can be incorporated with this model so that we can improvise the current model

10. REFERENCES

[1] Pravav P Nair, Ajay James and C Saravanan. "Malayalam Handwritten Character Recognition using Convolutional Neural Network" International Conference on Inventive Communication and Computational Technologies, October-2017

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "Imagenet classification with deep convolutional neural networks", In Advances in neural information processing systems, pages 1097–1105, 2012.

[3] Raju, Bindu S Moni, Madhu S. Nair, "A Novel Handwritten Character Recognition System Using Gradient Based Features and Run Length Count", Sadhana Indian Academy of Sciences , Springer India, 2014, p. 1-23

[4] Abdul Rahiman M and Rajasree M S, "An Efficient Character Recognition System for Handwritten Malayalam Characters Based on Intensity Variations", International Journal of Computer Theory and Engineering, Vol. 3, No. 3, June 2011

[5] Yann LeCun, Leon Bottou, Yoshua Bengio and Patric Haffner,"Gradient-Based Learning Applied to Document Recognition", PROC.OF THE IEEE, NOVEMBER 1998