CHAPTER 6

6. WRITER IDENTIFICATION MODEL THROUGH DEEP LEARNING

This chapter demonstrates the self-taught learning approach implemented in building the writer identification model using deep learning. In this work, Convolutional Neural Network (CNN) a kind of deep learning framework is employed. Self-taught learning attempts to capture subtle features based on training data. A Convolutional Neural Network technique learns multifarious and intellectual features automatically from handwritten text images and is employed in this research for identifying the writer. The modelling of writer identification task with self-extracted features through Convolutional Neural Network is described in this chapter. The experiment results are compared against the results of traditional Artificial Neural Networks and illustrated in this chapter with tables and charts.

6.1. MODEL V - WRITER IDENTIFICATION MODEL THROUGH CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks in deep learning have its own dimension to generate new features from a limited set of training dataset. CNN could be particularly advantageous for Tamil Handwriting Writer Identification for several reasons [85-86]. First, CNN carries out features learning and classification inside a unified framework. As the features are automatically learned from the data itself, it may be feasible to capture subtle features to isolate the puzzling characters seen in Tamil handwritings. Secondly, the performance of CNN is very good in extraction of high-level features [87-89]. Specifically, the convolution and subsampling Layers utilized by CNN have been indicated to be very efficient in dealing with shape changes that will probably be the key challenge in coping up with the too much of cursiveness in Tamil writings. Hence in this work CNN is adopted to develop the writer identification models using character, word and paragraph Tamil handwritten images. The process flow of CNN based model is depicted in Fig. 6.1.

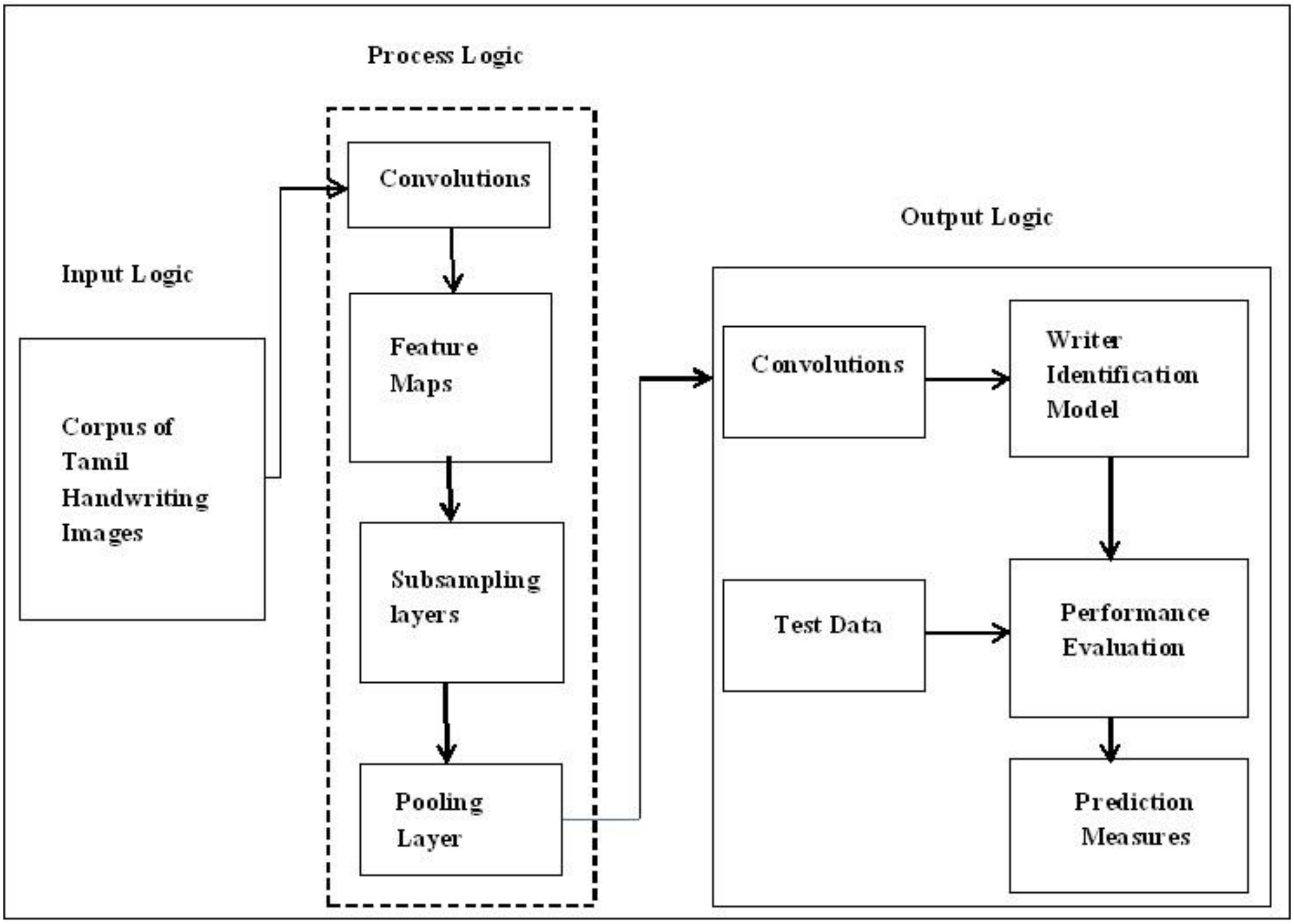

Fig. 6.1 Process Flow of CNN Based Writer Identification Model

Building the Model

The corpus of Tamil handwritten images prepared for the development of previous models is again used here in building CNN based writer identification model. Three corpuses of text images characters, words, paragraphs developed as described in chapter 3.2 are referred as CNNC, CNNW and CNNP respectively and used directly as input images for training the network. CNNs are implemented through number of interconnected layers. The text images are layered into number of repeated blocks of convolutional, ReLU and pooling layers. The convolutional layers convolve their input images with a set of filters. The filters are automatically learned during network training. The ReLU layer adds nonlinearity to the network, which enables the network to approximate the nonlinear mapping between image pixels and the features of an image. These learned features are, also known as activations, from one layer which will become the inputs for the next layer. Finally, the learned features become the inputs to the classification layer which uses softmax activation function at the end of the network. These self - learned features extracted automatically by the Convolutional Neural Network are more reliable for learning the classifier. Adam optimization is used to fine-tune the weights in order to reduce the error rate.

Considering both accuracy and efficiency of n-layered Deep Neural Network architecture, appropriate parameters such as batch size, epoch, input layer, subsampling layer and error rate are fine tuned for learning convolutional neural network. Convolutions with different batch sizes such as 5, 6, 10, with number of iterations such as 1, 5, 10, with input layers 5, 6, 10 and subsampling layers 10, 12, 20 are considered for learning the CNN classifier. The three datasets are partitioned into training and testing sets in the ratio of 80% and 20% and three independent CNN classifiers have been built.

Performance Evaluation

The performance of the three classifiers is evaluated with respect to accuracy, precision, recall and F-measure under different settings of hidden layers. The results of CNN classifiers with different batch sizes 5, 6, 10 and their epochs 1, 5, 10, input layers 5, 6, 10 and subsampling layers 10, 12, 20 are obtained for the test datasets and presented in Table 6.1.

Table 6.1 Results of CNN Based Models

Batch Size	Epoch	Input Layer	Subsampling Layer	Accuracy (%)	Error rates (%)	Precision	Recall	F-Measure
CNNC								
5	1	5	10	82	18	0.867	0.905	0.886
6	5	6	12	85.7	14.3	0.894	0.910	0.902
10	10	10	20	91.2	8.8	0.918	0.779	0.843
CNNW								
5	1	5	10	90	10	0.919	0.963	0.941
6	5	6	12	95.2	4.8	0.958	0.986	0.972
10	10	10	20	95.4	4.6	0.964	0.691	0.805
CNNP								
5	1	5	10	90.6	9.4	0.916	0.958	0.937
6	5	6	12	94.9	5.1	0.961	0.962	0.961
10	10	10	20	95.6	4.4	0.981	0.829	0.899

6.2. COMPARISION OF MODELS BASED ON ANN AND CNN

ANN is based on a collection of connected units or nodes called artificial neurons which loosely model the neurons in a biological brain [90]. Every connection is similar to the synapses that are present in the biological brain and transmits signals between the artificial neurons. An artificial neuron that is a recipient of the signal processes the same and furthermore confers the same signal to the other connected artificial neurons. Normally in ANN implementations signal connection that is existent between artificial neurons is the actual number, every single artificial neuron’s output is intended using the linear function of the sum of its inputs. The connections that exist between artificial neurons are referred to as the 'edges'. Generally both artificial neurons and edges possess a weight which is accommodated as the learning proceeds. This weight usually causes an increase or decrease in the connection’s signal strength. Typically, artificial neurons are aggregated into layers. Different layers may perform different kinds of transformations on their inputs. Signals travel from the first layer (the input layer), to the last layer (the output layer), possibly after traversing the layers (hidden) multiple times.

An experiment was carried out by implementing ANN with the same three datasets TNWIC, TNWIW, TNWIP (described in Chapter 4) containing handcrafted features by setting the number of hidden layers 4, 6, 8 and the models are evaluated with same metrics. The results of ANN classifiers obtained for the test datasets are presented in Table 6.2.

Table 6.2 Results of ANN Based Models

Hidden Layer	**Accuracy (%)**	**Precision**	**Recall**	**F-Measure**
TWINC Datasets				
4	72%	0.733	0.968	0.834
6	75%	0.752	0.895	0.818
8	79%	0.871	0.867	0.869
TWINW Datasets				
4	78%	0.745	0.702	0.731
6	80%	0.803	0.875	0.840
8	82%	0.816	0.934	0.853
TWINP Datasets				
4	81%	0.819	0.852	0.835
6	84%	0.830	0.757	0.792
8	86.5%	0.926	0.761	2.001

The results of recent CNN model are compared with traditional ANN against various performance metrics to validate the performance CNN based writer identification model. Artificial neural network implementation was done using the same three datasets described in chapter 4. The comparative results of both CNN and ANN based writer identification models are presented in Table 6.3.

Table 6.3 Performance Comparison of ANN and CNN

Datasets	Accuracy (%)	Precision	Recall	F-Measure
ANN				
TWINC (Character)	79%	0.871	0.867	0.869
TWINW (Word)	82%	0.836	0.937	0.883
TWINP (Paragraph)	86.5%	0.926	0.761	2.001
CNN				
CNNC (Character)	91.2	0.918	0.779	0.843
CNNW (Word)	95.4	0.964	0.691	0.805
CNNP (Paragraph)	95.6	0.981	0.829	0.899

CNN based writer identification models performs better when compared to ANN classifier for all three datasets. The proposed CNN classifier achieves 91.2%, 95.4% and 95.6% of accuracy for character, word and paragraph text image respectively in batch size 10. Also the error rates are less for CNN with 8.8%, 4.6% and 4.4% respectively in batch size 10. CNN achieved high performance of 95.6% compared with ANN of 86.5%. The recognition rates and error rates of CNN models for different handwriting text images measured with respect to batch size (epochs) along x axis and recognition rate (%) along y axis are depicted in Fig. 6.2 to Fig. 6.7. and the comparative analysis of ANN and CNN are illustrated in Fig. 6.8.

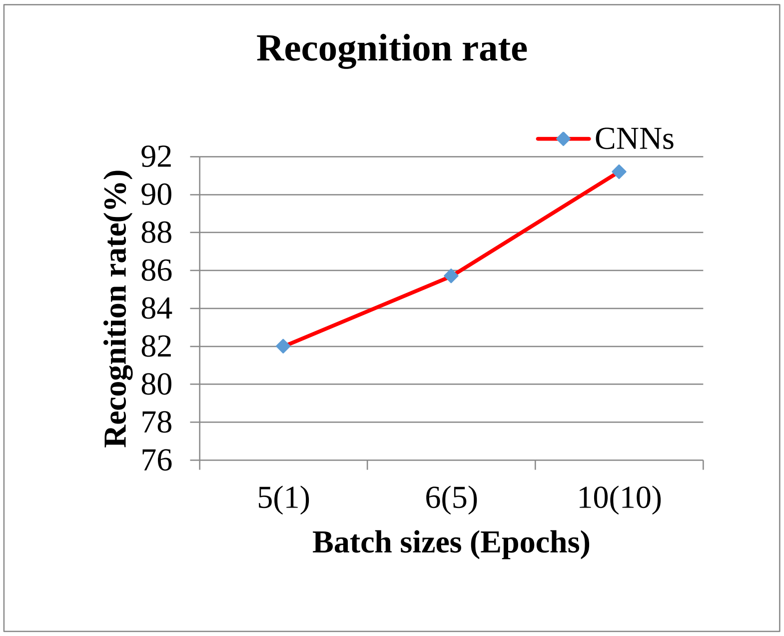

Fig. 6.2 Recognition Rate of Character Text Image Identification

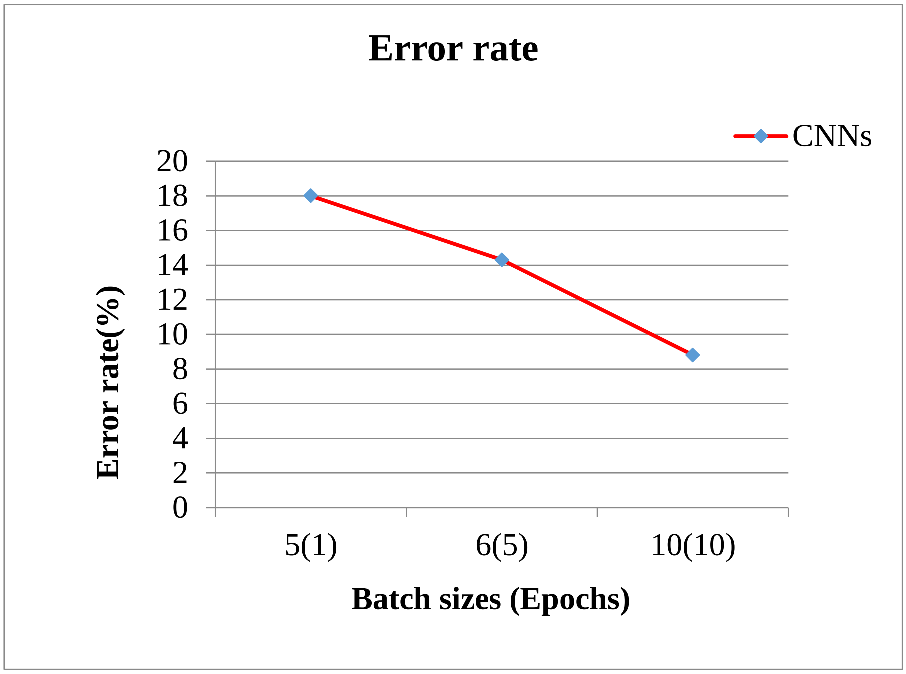

Fig. 6.3 Error Rate of Character Text Image Identification

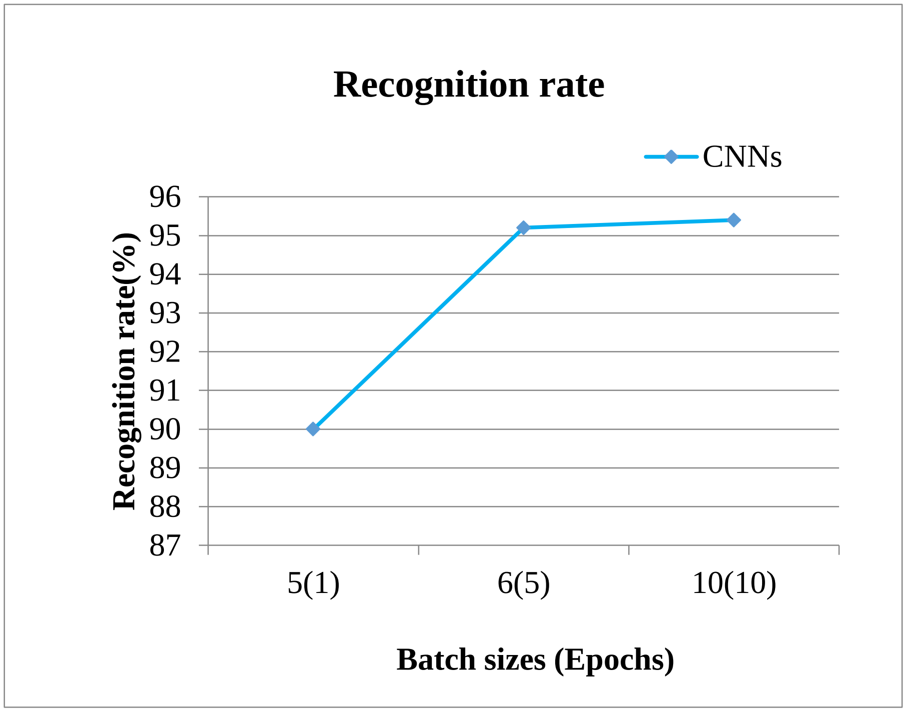

Fig. 6.4 Recognition Rate of Word Text Image Identification

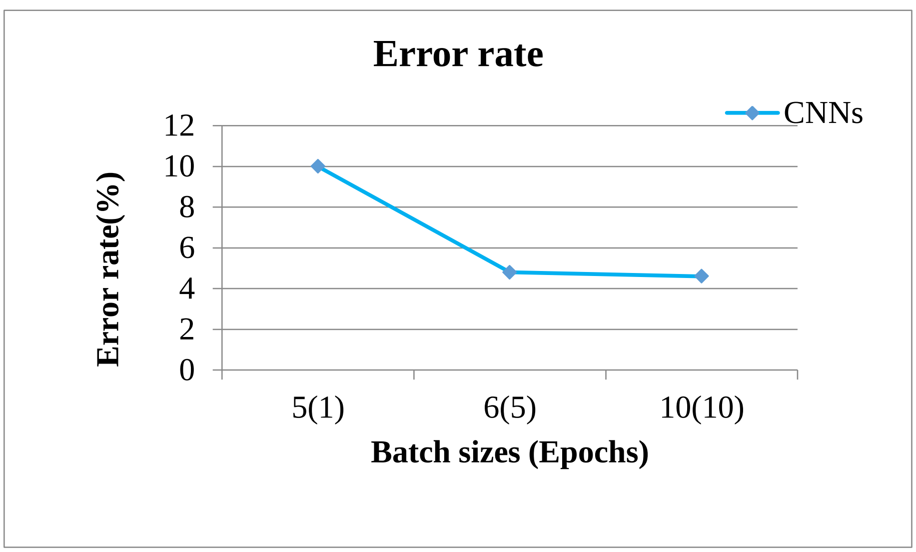

Fig. 6.5 Error Rate of Word Text Image Identification

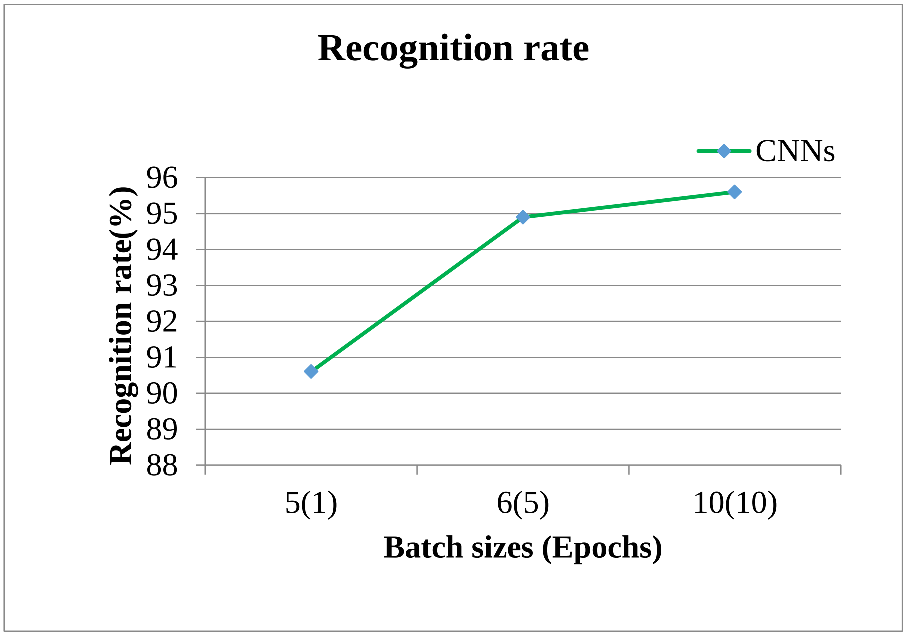

Fig. 6.6 Recognition Rate of Paragraph Text Image Identification

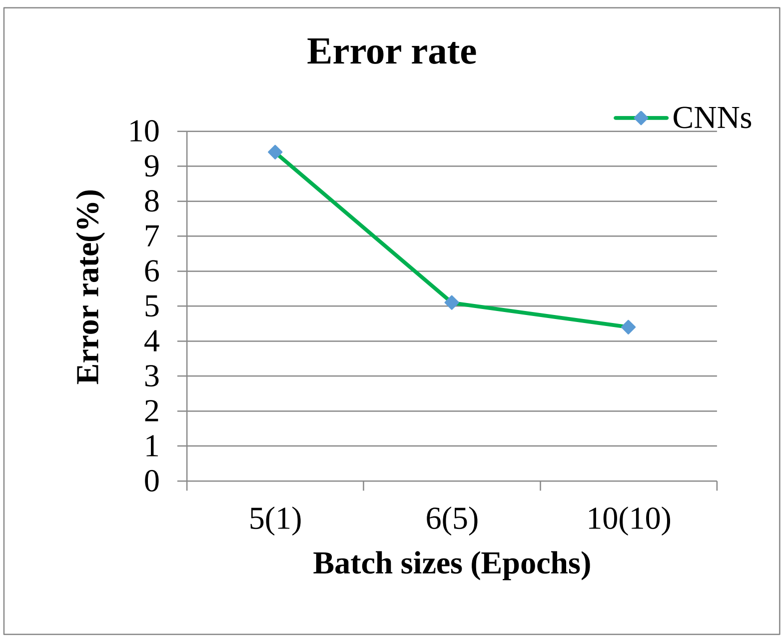

Fig. 6.7 Error Rate of Paragraph Text Image Identification

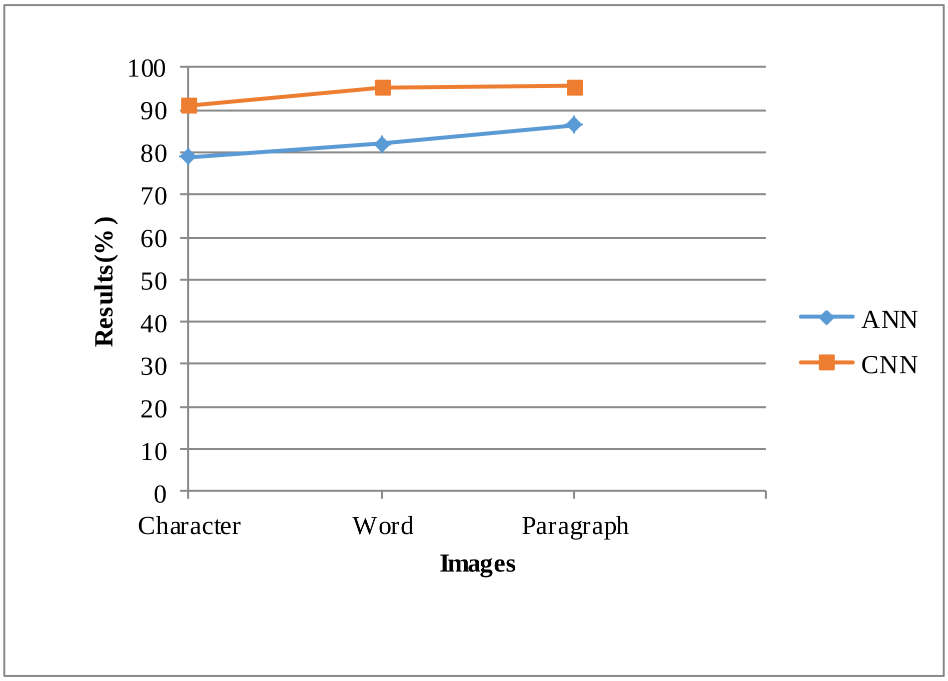

Fig. 6.8 Comparative Analysis of ANN and CNN

Findings

Through the results of this experiment it is observed that CNN can handle a complex problem of writer identification with good results. It is verified that the suitable choice of hyper-parameters significantly improves the prediction performance of the models. It is evidenced that CNN can learn complex, hidden and high level features from the input text images through several levels of subsampling and pooling layers. The recognition rates obtained by CNN establish that the models are very effective in absorbing shape variations of handwritten images. It suffers less from the high dimensional problem which minimizes the error rates of the prediction models. The unified framework of CNN enabled feature learning and classification within the deep learning environment which helps to reduce the number of tasks in building models through shallow learning and to predict writer more accurately.

6.3. SUMMARY

This chapter portrayed the application of Convolutional Neural Network in building data driven model for Tamil Writer identification. The experimental setup for implementation of CNN with various parameters is also discussed in this chapter. The results of CNN classifiers are presented in charts and tables and the findings are summarized. The exhaustive experiments developed in chapter 4, chapter 5 and chapter 6 encourage to easily upgrade the writer identification models for other Indian languages.