End To End Bangla Handwritten And Scene Text Detection Using Convolutional Neural Network

Inspiring Excellence

By

SOMANIA NUR MAHAL
13301124
B M ABIR
12201022
FAHIM BAKHTIAR
16341028

Department of Computer Science And Engineering
BRAC UNIVERSITY
Supervised by: AMITABHA CHAKRABARTY
Co-supervised by: Md Saiful Islam

Thesis report submitted to the BRAC University in accordance with the requirements of the degree of BACHELOR IN COMPUTER SCIENCE AND ENGINEERING in the Dept. of Engineering and Computer Science.

SUBMITTED ON: 21ST AUGUST 2017

Abstract

Handwritten text detection from a natural image has a large set of difficulties. A systematic approach that can automatically recognise text from handwriting, printed books, road signs and also classifies text and nontext blocks from natural image has many significant applications. For instance, visual assistance for visually impaired people, image understanding, classification of text in image, implementing autonomous navigation system. Recent development of deep learning approach has strong capabilities to extract high level feature from a kernel(patch) of an Image. In this thesis we will demonstrate an alternate approach that integrates a multilayer convolutional neural network (CNN) with supervised feature learning .This approach allows a higher recall rate for the text in an image and thus increases the overall performances of the system. And we have used these methodologies to create a learning model using synthetic and real-world data that is capable to process bangla and english handwritten and scene text in natural image.

DEDICATION AND ACKNOWLEDGEMENTS

We would like to share our heartfelt gratitude to our honorable Supervisor and Co-Supervisor Dr. Amitabha Chakrabarty and Saiful Islam respectively for their immense support and motivation for our work. We were inspired by them to learn and enrich our knowledge for this complex thesis project. Then we would like to thank our family for taking good care of us during this period of time and providing an environment of peace and harmony which was essential for every research efforts.

In Addition, we would like to thank the team behind the BanglaLekha Isolated[6] from University of Liberal Arts Bangladesh for providing the Bangla handwritten character dataset which had a tremendous impact on our project.

Author's declaration

We, hereby declare that this thesis is based on the results found by ourselves. Materials of work found by other researcher are mentioned by reference. This thesis, neither in whole or in part, has been previously submitted for any degree.

SIGNATURE OF THE AUTHORS:

.....
B M Abir - *12201022*

.....
Somania Nur Mahal - *13301124*

.....
Fahim Bakhtiar - *16341028*

SIGNATURE OF Supervisor:

.....
DR. AMITABHA CHAKRABARTY
ASSISTANT PROFESSOR, DEPT. OF COMPUTER SCIENCE and Engineering
BRAC UNIVERSITY

SIGNATURE OF CO-Supervisor:

.....
Md. Saiful Islam
LECTURER , DEPT. OF COMPUTER SCIENCE and Engineering
BRAC UNIVERSITY

TABLE OF CONTENTS

List of Tables

Table **Page**

List of Figures

Figure **Page**

CHAPTER 1

INTRODUCTION

In current stage of our world, information is considered to be one of the most significant driving force for our social change.[1] And language is the vessel through which information flows. But it seems that there is a clear division between digital and printed or handwritten text that surrounds us. To bridge that gap, computer vision researchers have gone to great lengths such as creating mathematical formulas and using the techniques of hand-engineered features[1], whereas the deep and detailed knowledge of the feature is incorporated into models to process the image. So that they can detect and recognize characters , numbers and even symbols. But formulation and experimentation of each and every filters and feature for each and every characters, and take account for other various factors, is hard and tedious task.

1.1 Motivation

Extracting text using end-to-end image recognition systems have received a lot of attention as it allows the system to be robust and the use of such models in variable real world Scenarios.The field of computer vision has seen great strides towards expert systems that can be trained using large image dataset due to the rise of high performance computing(HPC) systems. High performing parallel processing GPUs has enabled the computer vision researchers to turn the problem of image recognition into a problem of numerical optimizations problem. So now there exists a wide range of algorithms and architecture to tackle such problems. One such architecture to create models and solve those problems are neural networks. It has been designed on the basis such that it imitates some function of the neural pathways of human brain. And to maximize the performance of such architecture, numerous methodologies have been proposed and one such implementation of a system of detector and recognizer that uses machine learning techniques is convolutional neural network or CNN [3], [4],[5]. In CNN, the state of the art detection

performance is 80 percent on F-measure with ICDAR 2011[5]. And for end-to-end recognition, the result is 57 percent accurate on SVT dataset[6].But even then, the accuracy of such algorithms like used in common OCR system is not up-to the level of human level accuracy. They are clearly focused on document text detection. Those systems are also plagued by different lighting conditions of the images and various background noise makes the text recognition systems to under perform [2].

1.2 Goals

We aim to construct a system that will be put into use in various useful applications. To name some, text and non-text segregation, image understanding, an alternative approach to OCR technology, and more. Even though works have been done previously, on character recognition via neural networks, trying to employ it with Bengali handwritten text will put on a big milestone. Doing it with efficiency and high accuracy is where our main focus will be lying on, in order to achieve the goal we set for this project.

1.3 Objectives

In this paper we will be experimenting on different types of convolutional neural network architecture which will be appropriately evaluated on the context of different english alphabet datasets like ICDAR 2011 and bangla alphabet dataset like ISI Bangla Scene Character Database[7] for scene text recognition . And for Handwritten dataset, we will also be analysing the accuracy performance using english alphabet datasets like UJI pen character dataset[8] and bangla alphabet datasets like BanglaLekha-isolated [9]. By analysing these different datasets we will establish a base ground truth for a superior CNN architecture that can be used in both handwritten and scene text detection in Bangla and English text recognition. And ultimately with the chosen CNN architecture we will implement an end-to-end multilingual character recognizer that will be able to localize and recognize text from any background image. With the experimentation results we determine which architecture is performing better for our current problem.

CHAPTER 2

BACKGROUND STUDY AND RELATED WORKS

End-to-end text recognition in natural images has two key components (i)text spotting and (ii)text recognizing[23]. The main objective is to locate the specific text oriented regions. of text then recognize the actual words.Many research and work have been done for solving the character recognition problem[20][23]. Much approach and methods are implemented and showed higher performance.However in a complex image locating and recognizing character still remains a problem. Real time text localization in an image has large computational cost [22], sliding window[6][2][7] and connected component approach.

2.1 Sliding Window Approach

Sliding-window method work such a way where a multiscale sub-window is sliding through a whole image for detecting possible locations of text[8][4][2][12][13][14][15][16][17]. This method limits the search to a subset of image rectangles thus this reduces the number of subset checking for text.A pre-trained classifier is used for identifying presence of text within sub-window.The approach of wang, et al. [7]locate each character as visual words using sliding-window method where a Random Ferns classifier[2] is used with a histogram of oriented gradients (HOG) feature.Pan et al.[9] also used sliding-window method for generating text confident map WaldBoost [19] and HOG feature.Maintaining computational flexibility and designing a discriminative feature to train a powerful classifier are the main challenges of this group of method[3].

2.2 connected component approach

As connected component approach can separate text and nontext components with a complexity of O(N)[22][3] [5][1][6][12][15][14][16][18] [5] [10] [8] [21] [1] [17] .Thus this method has

higher performance rate in text detection and localization[3].A fast low level detector is used for separating text and nontext components at pixel level were retained pixels are grouped together to form possible text information[1].Stroke Width Transformation and ERs/MERs [11] [13] approaches are used in connected components method.In [1] implemented a CE-MSERs detector which is capable of identifying complex and ambiguous image components.This approach extends conventionally used MSERs in such a way that enlarges the local contrast between text patterns and background[1].This CE-MSERs detector integrated with a text-CNN filter to implement a text detection system.This system attains ICDAR 2011 and ICDAR 2013 benchmarks on state-of-art result. In[4[32]] [23] proposed a system which allows extensive pre-processing stages using multi-stage pipeline with incorporation of connected component analysis via a conditional random field(CRF).Huang et. al. [15]extends original SWT and proposed a Stroke Feature Transformation(SFT) .For tracking pixel this proposed system integrates important color cues of text patterns.

2.3 Machine Learning Methods

A wide range of approaches are used for text segmentation and recognition. In an end-to-end system text detection system identify the text region which is the input of character or word recognizer[23] .Then the recognizer recognizes each character in the word or the whole word. Deep convolutional neural network models has higher performance rate for recognizing and computing high level deep features [38][39][40][41]. LeNet a traditional CNN model, powerful for digit and hand-written character recognition[42][43]. Advancement of deep CNN models help to solve challenging scene text recognition [1],[4] ,[6][2],[3].A Fully-Convolutional Regression Network implemented by Gupta et al.that integrates the both text detection and bounding-box regression feature[44].Deep neural models are significantly improve the performance than previous manually featured design.

CHAPTER 3

METHODOLOGY

Main objective of our work is to build a bangla character recognizer which can automatically locate and recognize character from any simple or complex image components. For building this system we are going to analyze three architecture using supervised learning. These three approaches are fully connected Stochastic gradient descent and fully connected one layer hidden deep neural network using rectified linear unit (RELU) activation function and Multilayer Convolutional Inception Module. The performance analysis of these three architectures are calculated and the one with higher accuracy is applied for building the system. In the following sections we present the details of each approach that we followed and the result of our experiments. After discussing about the experimental result a performance analysis is presented for the each architecture.

3.1 Fully Connected Stochastic Gradient Descent

3.1.1 Logistic Regression

Classifying a large and complex set of data makes the task of mathematical models a daunting one. Character recognitions main complexity arises from the task of classifying several classes of characters. For the problem of classifying image into few classes make the computation even more challenging. So using functions such as linear regression is not an option, as it will not be properly extract the feature of several classes of characters. So in order to properly model the dataset into a mathematical formula, there have been usage of multinomial logistic regression. It is specially used in case of the models whose complexity is polynomial in nature.

$$\text{logit(y=1)} = \log\left(\frac{p(y=1)}{1-(p=1)}\right) = \beta_0 + \beta_1 \cdot x_{i2} + \beta_2 \cdot x_{i2} + \ldots + \beta_p \cdot x_{in} \text{ for } i = 1 \ldots n\,.$$

This formula or commonly known as logits, is used to predict the label or score (Y) of a character whose image is given into the system as an input of X. Input images or features are provided as a matrix form. We have to find the weights and biases while we are training, which is to be found by training. This weights and biases are the coefficient which is to be used to calculate the score when the training of the model is complete.[1]

3.1.2 Cross Entropy

To train a model, our architecture will tune the parameters like the weights and biases on the logits function. In order for our system to determine how much of the weights and biases value to be changed in order to improve the accuracy of the prediction, we will calculate the loss function. To calculate the loss function we will use the cross entropy formula. Cross entropy is a method of comparing the scores predicted from each training or testing step and comparing it with the the ground truth label from the dataset. It is considered to be the distance between the predicted scores and Labels.

$$L(s,l) = -Li.Log(Si)$$

Or Simply Loss(score,label)=-Labels.Log(Scores)

3.1.3 Stochastic Gradient descent

In order to train the parameters of the logits, an optimizer is used to reduce the loss function of the model in a particular training instance using a technique called gradient descent optimization. First, the gradient of the score is calculated using the loss function and objective function.

$$\Theta = \Theta - \eta * \bigtriangledown(L(s,l),\Theta)$$

J()is the objective function which considered to be the goal of the gradient descent algorithm. The gradient descent algorithm will try to reduce the loss function by comparing it with the objective function. In order to reduce the loss function, it will change the weights and biases of the logits on each training step. But the gradient descent algorithm is only suitable to reduce the loss function only when the training dataset is small. This is because the training time for systems using gradient descent increases rapidly when the data set size increases. So in order to use large datasets we have followed a modified version of gradient Descent called stochastic gradient descent algorithm. The main difference is that the dataset is randomly shuffled and a minibatch of data is picked that is to be used to train the model.

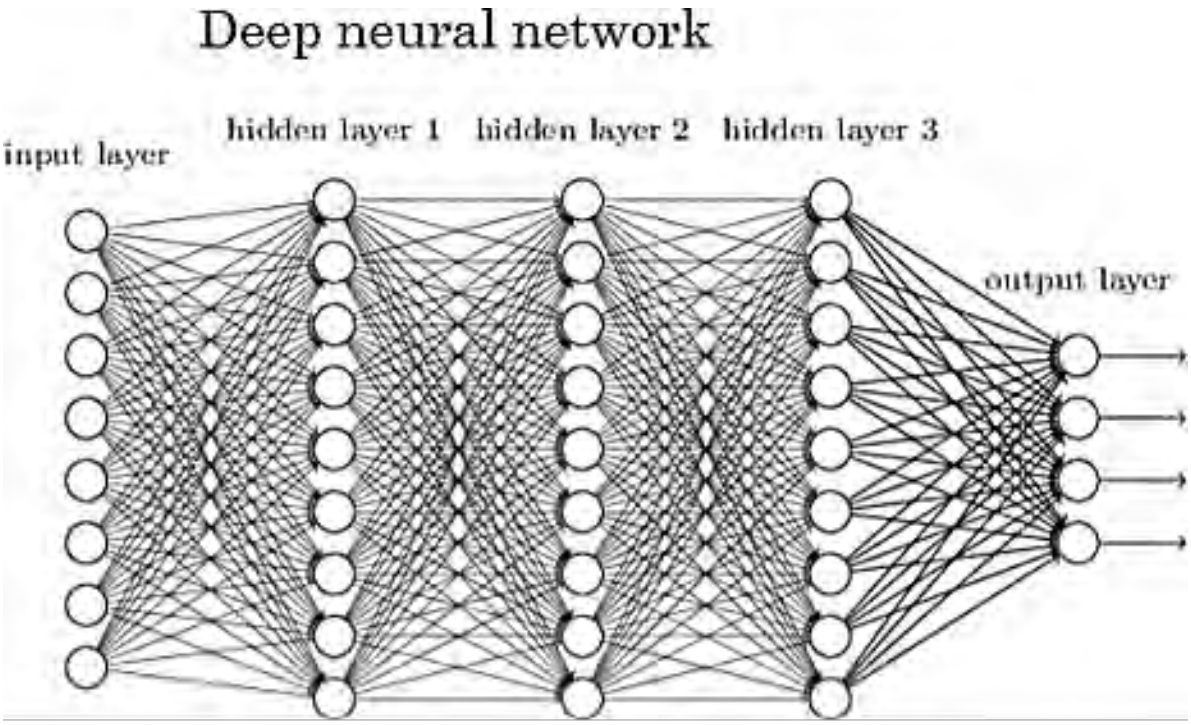

FIGURE 3.1. Fully Connected Neural Network

3.2 Deep Neural Network

The above described architecture is considered to be a single layer neural network model. But to increase their accuracy, a deep neural network model is used. In deep neural network, output score of a single logits layer is used as an input to another layer using the different activation function.In our model we used 1 layer hidden deep neural using rectified linear unit activation function.The hidden layers breaking down the image components so that it can decode the whole image.One layer hidden neural network contains a single Relu or hidden layer

3.2.1 Relu or Hidden Layer

Rectified linear unit ReLU is a activation function which receives the output of logits or neuron and filters it to speed up the training process as connecting two layer slows down the training process. It converts the linear output of one layer into a non-linear function.

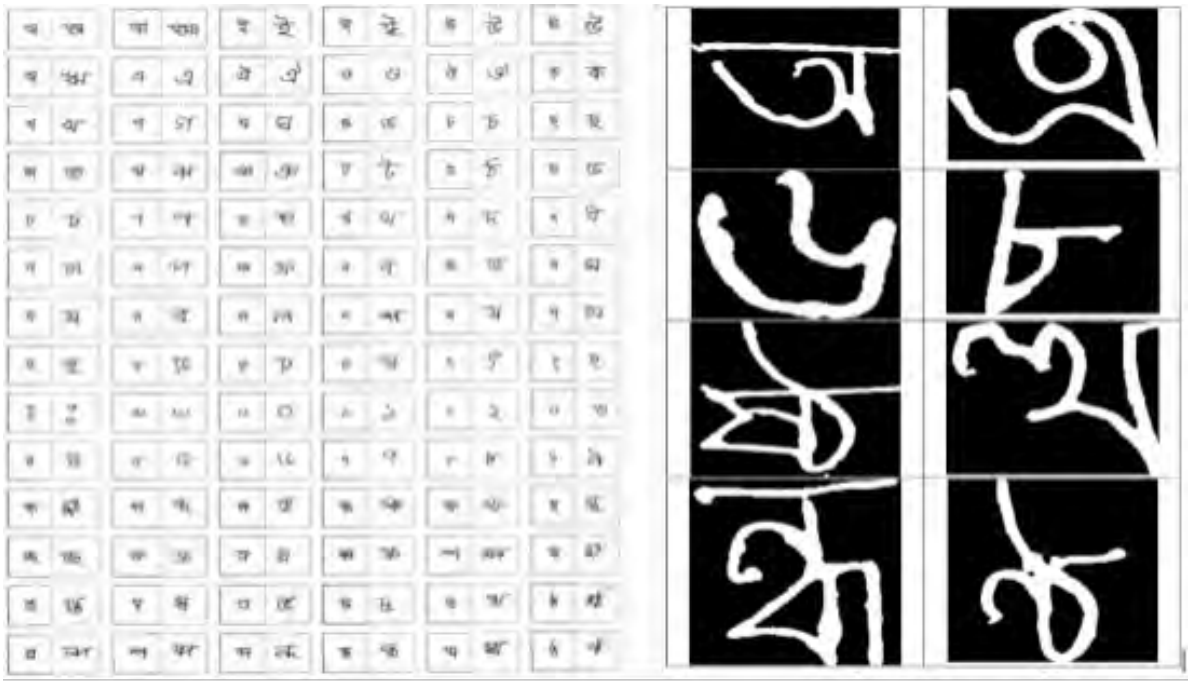

FIGURE 3.2. Samples of Bangla HandWritten Characters

3.3 Recognition Using CNN And Inception Module

In this section our proposed multilayer inception based convolutional model explains in detail. Following subsection gives a brief idea of dataset, image preprocessing, system architecture and classifier.

3.3.1 Dataset

The text recognition subsystem is trained on character-level by using BanglaLekha-isolated [6] dataset. BanglaLekha-isolated has total 1, 66,105 handwritten character images. It consists sample of 84 different Bangla handwritten numerals, basic characters and compound characters [6] which collected from different geographical location of Bangladesh and different age group. In fig. 3.2 collected from [6], a sample of Bangla handwritten character of a particular age is shown. Where first 11 characters are vowel followed by 39 consonant characters then 10 characters are Bangla numerals and rest 24 are Bangla compound character. The dataset provides multiple labels per character/character group. It contains a wide variation of frequently used in Bangla compound sentences which are very complex shaped. In fig. 3.2 last two character are compound character.

```
  -0.5 -0.5  0.5  0.5  0.5  0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5  0.5 -0.5  0.5  0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5  0.5  0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5  0.5  0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5 -0.5  0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]
 [-0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5
  -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5]]
```

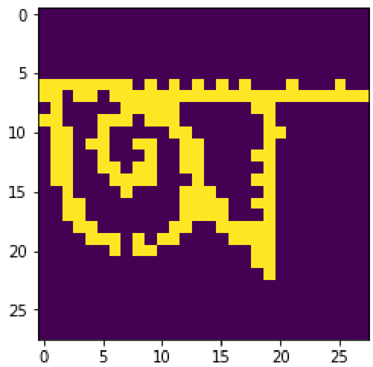

FIGURE 3.3. Converting Images into numpy Array

3.3.2 Image Preprocessing

As these data from dataset has different image size so it was a bit challenging to train them as we used fixed image size which is 28*28 pixel for each image. Therefore image pre-processing is needed for handling this big size of data. For resizing the images we used python imaging library PIL. Anti-Aliasing filter is used so that quality of the image doesn't degrade while resizing. After setup the train environment we convert images into numPy and save these images in separate class file. Then images of each class are shuffled to have random validation and training set. Also merge them into a single dataset. Duplicate data between training and test can skew the result. So, we need to remove all the overlap between training and test data. We measured the duplicates data and find 1081 samples overlap between valid and train data, 1278 samples overlap between test and train data, 185 samples overlap between test and valid data. Then we create a sanitized test and valid datasets. This data cleaning process gives better accuracy.

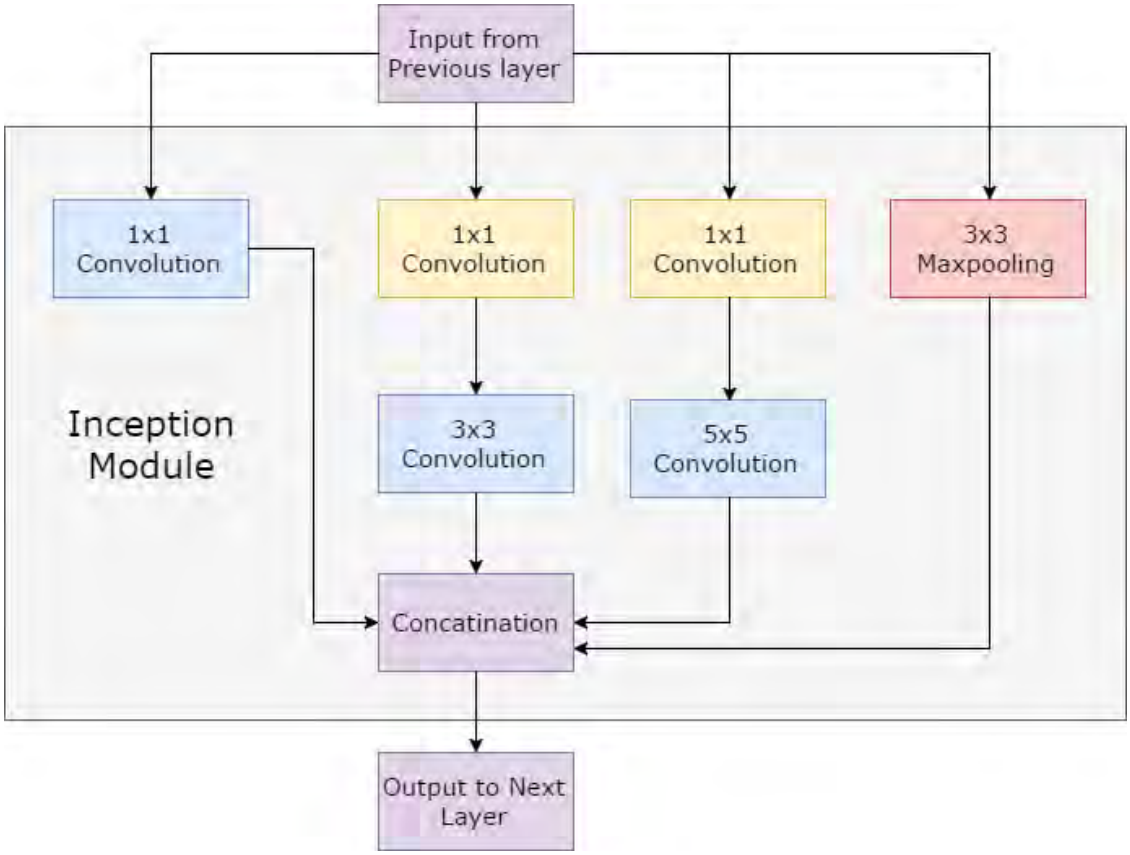

FIGURE 3.4. A detailed Inception Module Architecture

3.3.3 Model Architecture

Our system architecture consists of several layers of Convolutional Neural Network. As shown in fig.3.4, the pre-processed data with a size of 28x28 image is taken as an input into the 1st convolutional neural network layer with patch or kernel size of 3x3 and 'same' padding property. After Max Pooling, the output is again fed into a 2nd convolution layer with patch size of 5x5 and 'same' padding. Its output again fed into Max Pooling layer of same configuration as before. The max pooling from the previous layer is used as the input of the Inception module, where CNN is not only stacked sequentially but also parallel. Then output of the inception module is fed into a fully connected network where every single node is connected with each other. This layer flattens the high level feature outputs from the inception module and converts it into a nonlinear combination of these features. After that, another fully connected layer is added to flatten the data again.

3.3.4 Classification Using CNN and Inception Module

Classifying Bangla handwritten character has high –dimensional complexity which requires a multilayer, hierarchical neural network [10]. For visual recognition three principle factors are very important for instance, local receptive fields, weight sharing, and subsampling layer which makes a difference in CNN from other simple feed forward neural network[3].In our proposed CNN based approach we used two convolution layer followed by one inception module. Unlike other traditional MLP based method, here CNN itself capture local features from the input image by forcing each filter of CNN layer depends on spatially local patch of the previous layer. Therefore a feature map is generated in next layer. In CNN a great advantage of weight sharing is it significantly reduces the number of free parameter [3] by sharing the same set of edge weights across all features in the hidden layers. Subsampling or pooling layer is another powerful building block of CNN. The main objective of pooling layer is to reduce the dimensionality of the response map created by convolution layer [3]. And also allows translation invariance for instance, rotation, scale and other distortions into the model. In our approach we used an inception module on top of convolutional layers. This module works as filter inputs of multiple convolution which works on same input. For instance, a 1x1, 3x3, and 5x5 convolution layer works on same input and also employ pooling layer at the same time. This module improves the performance of overall model by extracting multi-level feature.
In fig.3.5 shows structure of convolutional neural network used for Bangla character recognition. Two convolutional layer with 3x3 and 5x5 receptive fields along with two max pooling and ReLu layer. For instance, a 28x28 pixels image is input to the CNN. Applying 1st CNN layer on input produced first level feature maps. These feature maps are produced such a way that it can extract a variation of local features where distinct patches with different weights and biases from other patches are used.

After that a max-pooling layer down-sample the input (output matrix from previous layer) representation and also reduce the dimensionality. Input 28x28 pixel image down-sampled to 14x14 feature maps after max-pool operation. The 2nd layer CNN and max-pooling operations are same as 1st layer convolution. Output of 2nd layer CNN is input of inception module. In this inception module 1x1, 3x3, 5x5 convolution along with 3x3 max-pooling is used. Here, 1x1 convolution reduces the dimensionality of the input to large convolutions. Therefore it makes the computations reasonable. Filters from all layers in inception module concatenated and converge the output with two fully connected layer. And a dropout activation function is employed to maximize the performance.

3.4 Character Localization

In order to process words we have taken the approach of breaking it down into character components and then providing that each character into our trained character recognition model

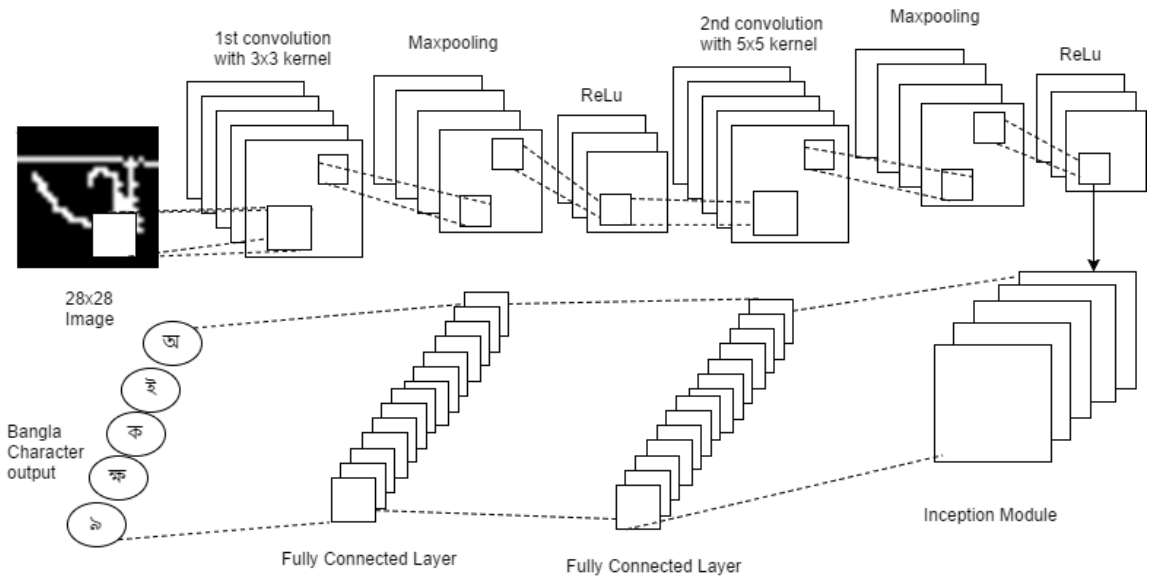

FIGURE 3.5. Data Flow between the proposed CNN Architecture for Bangla Character Recognition

to give the prediction. To break down the word we have used MSER algorithm which detects edges in an image and suggests it to our recognition model. The basic idea of Maximally Stable Extremal Regions is to detect blobs(distinguished region) where nearby pixels are grouped together in terms of similar intensity and color. And after that they try to maximize the stable region. The algorithm recognize contiguous sets of pixels where intensity of outer boundary pixels are higher than inner boundary pixel intensity.So these regions are considered as maximally stable region if they don't change much over a varying amount of intensities.For character localization we choose mser for our system because runtime complexity is so light and its

$$O(n * log(log(n)))$$

where n denotes total number of pixels on a image. As we processed all our images in real-time so mser is advantageous for system as it facilitates robust to blur and scale.

CHAPTER 4

EXPERIMENT AND PERFORMANCE ANALYSIS

In this chapter we evaluate accuracy of each of the approaches individually.Then our system is tested in all of these approaches with ISI Bangla Scene Character Database[7],UJI pen character dataset[8], BanglaLekha-isolated [9].We then analyze the result accuracy of each of the approaches to select the best fit approach for our system.

4.1 Classifier Using Logistic Regression

Firstly we train a model with logistic regression.Here 20,000 samples are used for train this model.After calculated the logistic we fit the model by logistic fit function and tuning different parameter to get better accuracy.We got 13 percent accuracy score, 13 percent precision score and 13 percent recall score for 500 sample. Only for one sample we got 90 percent fit score.

4.2 Fully Connected Network Using Stochastic Gradient Descent

To design a fully connected deep neural network model firstly we train a multinomial logistic regression using gradient descent. Here we use TensorFlow computational engine for softmax calculation.As gradient descent training takes longer time so we need to subset the train data for faster training.For computation TensorFlow is used where we need to give all input variables and the formula of our desired computation. Then these created as nodes over a computation graph.To train this model 10,000 train subset are used and weight matrix are initialised using random values with normal distribution. Also the biases are initialised to zero. After computing softmax and cross-entropy we take the average of cross-entropy to use gradient descent for

Table 4.1: Training and testing accuracy for different Architectures

	Linear Regression	**Fully Connected SGD**	**1 Hidden layer**
Training Accuracy	13.00%	82.20%	89.10%
Testing Accuracy	13.26%	89.10%	92.50%
Traing Time (seconds)	300	900	14400
No of Steps	500(samples)	300001	300001
Loss	0.8	0.55	0.33

minimizing loss. Here we got 78.7 percent training accuracy, 83.0 percent test accuracy. Then we used stochastic gradient descent in similar graph because it is more faster.In SGD we didn't use a constant node to store the train data instead we created a placeholder node which is fed actual data.We used 128 minibatch and run 30,001 steps. We obtained 82.2 percent training accuracy and 89.10 percent test accuracy for this model.

4.3 One Layer Hidden Deep Neural Network Using ReLU Activation Function

Now we turn the model into a 1-hidden layer deep neural network using rectified linear units where we used 1024 hidden nodes.And its improve the test and validation accuracy.We got 89.10 percent training accuracy and 92.50 percent test accuracy which is a great improvement. The learning rate of this model was set at 0.5.

4.4 Performance comparison between three Layers

The table 4.1 shows the performance comparison between different neural network architectures. The training and testing accuracy of linear regression is 13.00 percent and 13.26percent which is very low. The system was trained using 500 samples of image data from different classes and it took 300 second to So in order to increase the accuracy of the system we have to change the architecture from ground up. So we implemented a fully connected multinomial logistic regression with stochastic gradient descent architecture. With this implementation, the training and test accuracy increased drastically to 82.20 percent and 89.10 percent respectively. It took 900 seconds to train it with 30,001 steps or epoch.

The figure 4.1 illustrates the training accuracy of Multinomial Logistic Regression with Stochastic gradient descent(Red line) and 1 Hidden layer ReLU (Blue Line) architecture. The graph shows that the accuracy increases dramatically for Hidden layer with ReLU. From here we can infer that 20 thousand steps training is sufficient for the network to reach convergence.

The figure 4.2(a) shows the frequency of the accuracy for 1 Hidden layer ReLU (Blue Line) architecture, visualizing that the accuracy frequency is the highest for the accuracy of in range

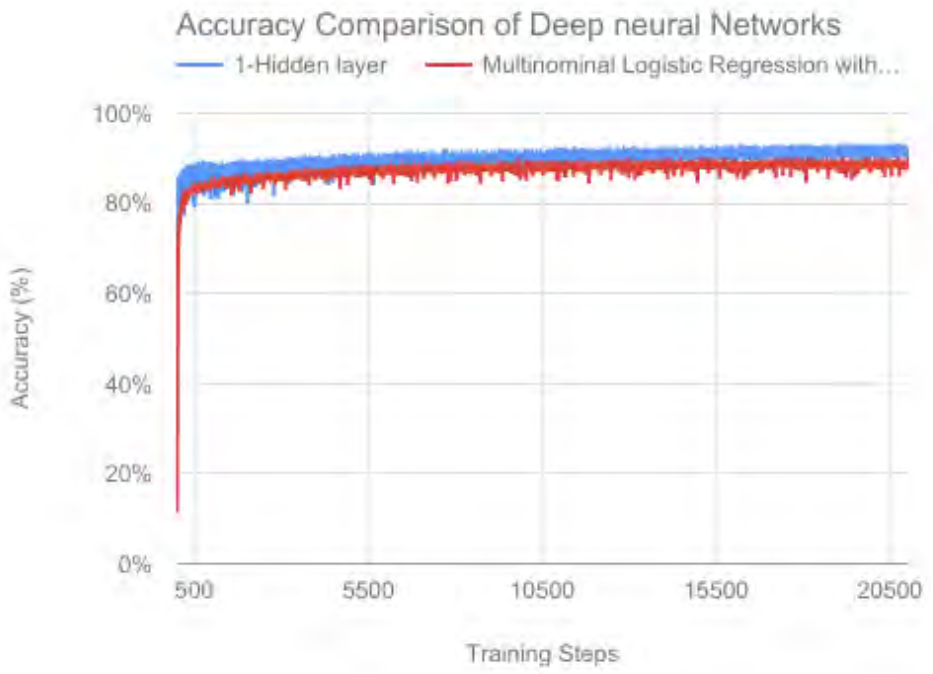

FIGURE 4.1. Training Accuracy vs Number of Training Steps

of 88.65 and 92.91. And figure 4.1(b) shows the frequency of the accuracy for Multinominal Logistic Regression with Stochastic gradient descent architecture, visualizing that the accuracy frequency is the highest for the accuracy of 89.36. After accuracy comparison we can conclude that Multinominal Logistic regression with SGD performs a more consistent accuracy for training the models.

(a) (b)

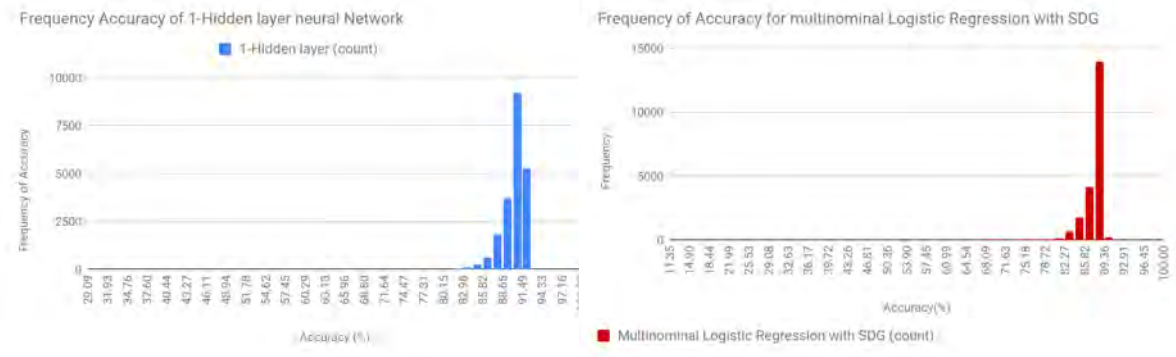

FIGURE 4.2. Frequency of Training Accuracy of (a) 1-Hidden layer ReLU (b) Multinominal Logistic Regression

Table 4.2: Variable hidden layer experiments

Hidden Layer	Steps	Mini-Batch Accuracy	Validation Accuracy	Test Accuracy	Minibatch Loss
128	30000	88.00%	89.80%	87.60%	0.587701
384	30000	90.00%	92.70%	88.60%	0.583768
512	30000	94.00%	92.60%	88.80%	0.503671
720	30000	88.00%	94.20%	**89.30**%	**0.397282**
750	30000	88.00%	**94.70**%	88.70%	0.348352
800	30000	94.00%	94.20%	88.20%	0.568419
900	30000	**98.00**%	93.80%	89.10%	0.345489

4.5 Our proposed architecture with hyper parameter tuning

We performed a through comparison on different hyper parameters that can be tuned, to measure the performance of our proposed architecture for Bangla handwritten character recognizer. We started with 128 hidden layers and 0.005 learning rate and got a decent mini-batch, validation and test accuracy of 88.0 percent, 89.80 percent and 87.60 percent respectively as shown in Table 2. One of the two main hyper parameters that needs most attention are numbers of hidden layers in the subsampling layer or ReLu layer that is found immediately after the convolution layer. Therefore, we intuitively changed the number of hidden layers and trained for the optimum test accuracy. It seems that increasing the hidden layers up to 720 increases the test accuracy result. Therefore with 900 hidden layer hyper parameter, we tested different variations of the learning rate hyperparameter, obtaining the results in Table 3. We started with a high learning rate of 0.055 and it provides a very low testing accuracy compared with the previous value of 0.005 learning rate. Therefore, we significantly decreased the learning rate and found a better testing accuracy with learning rate of 0.006 and best of the class validation accuracy. But increasing the learning rate more than 0.005 yields no more better results. The Fig 4.3 illustrates the result of increase in validation accuracy with number of steps. The validation accuracy increases sharply around 5000 Steps. And then it increases steadily afterwards. The final validation accuracy is around 89.10 percent with 900 hidden layers and 0.005 learning rate. Therefore it is apparent that the whole network has converged with 900 hidden layers and 0.005 learning rate for BanglaLekha dataset with 84 class Bangla character.

Table 4.4 shows comparison of test accuracy among proposed method, CNN based BHCH-CNN [10] and unsupervised deep belief network. The most significant part of BHCR-CNN and proposed method is that no feature selection techniques are used here. In [10] 50 classes (only Bangla basic characters) are only used in classification and the classification gives 85.96 percent

Table 4.3: Variable Learning Rate Experiments

Learning Rate	Steps	Mini-batch Accuracy	Validation Accuracy	Test Accuracy	Minibatch Loss
0.055	30000	84.00%	93.60%	88.40%	0.480084
0.006	30000	92.00%	**94.70**%	88.80%	0.305358
0.005	**30000**	**98.00**%	93.80%	**89.10**%	**0.345489**
0.003	30000	92.00%	94.60%	88.05	0.367663

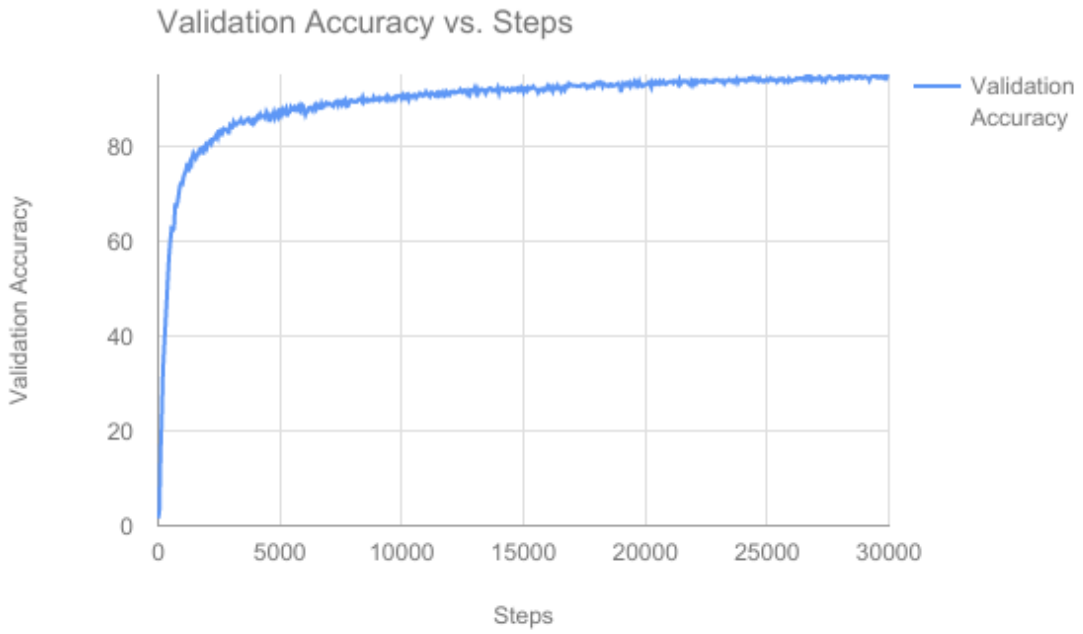

FIGURE 4.3. Validation accuracy vs training steps

recognition accuracy. In [11] an unsupervised deep belief network is used where 60 classes (10 Bangla numerals and 50 basic character) are used and have 90.27 percent recognition accuracy. No compound character are used for training the classification. In our proposed method 84 classes are used which consists of Bangla basic character, numerals and compound characters. Compound characters are complex in shape so it's a much challenging for a classifier to recognize compound character. Our proposed method successfully classify commonly used compound Bangla characters. Therefore our proposed classifier, with simple convolutional and inception network shows overall improved performance recognizing more classes. This shows the power of convolutional in generalizing a large number of handwritten classes and further improvement of recognition is possible with deep convolutional networks.

Table 4.4: Performance comparison with another CNN based approach and unsupervised approach

Other works	**Number of class**	**Classification**	**Compound character**	**Accuracy**
BHCR-CNN [10]	50	CNN	No	85.96%
DBNs[11]	60	Unsupervised Deep belief Network	No	90.27%
Our Proposed method	84	CNN with inception	yes	89.10%

CHAPTER 5

DEMONSTRATION

In this chapter we are going to show some practical output from our proposed system.These output are obtained by feeding our system some of the real world input that it would be faced if we deploy it to as an application.The system mainly consists of two part, first we will demonstrate the Bangla character recognition and then we will show the result generated by our Bangla character localization engine.

5.1 Character Recognition

We have developed a web graphical user interface where the user can draw a bangla character on the a canvas and captures that character as a image and send it to a backend flask application where it predicts the character which has been drawn on the canvas.The canvas implementation was done using HTML,CSS and JavaScript. The flask app loads the trained bangla recognition CNN model from saved files and use it to predict the character output.The trained model's graph architecture was reloaded from model.ckpt.meta file and the value for the models weights vaules are retrived from model.cktp.data file. This models files were generated while we were training the model using the training dataset. So in here the flask app is acting as a server which responds to the user input from the HTML view and returns the data using JQuery. When the user drawing is finished, he can submit the drawn image using the predict button and that's then the flask app calls the predict function in python definition and passes the image to the graph as an input to give an inference of that input. When the graph gives out the inference as a single class of bangla character, the flask app then send the output result to the HTML using JQuery.

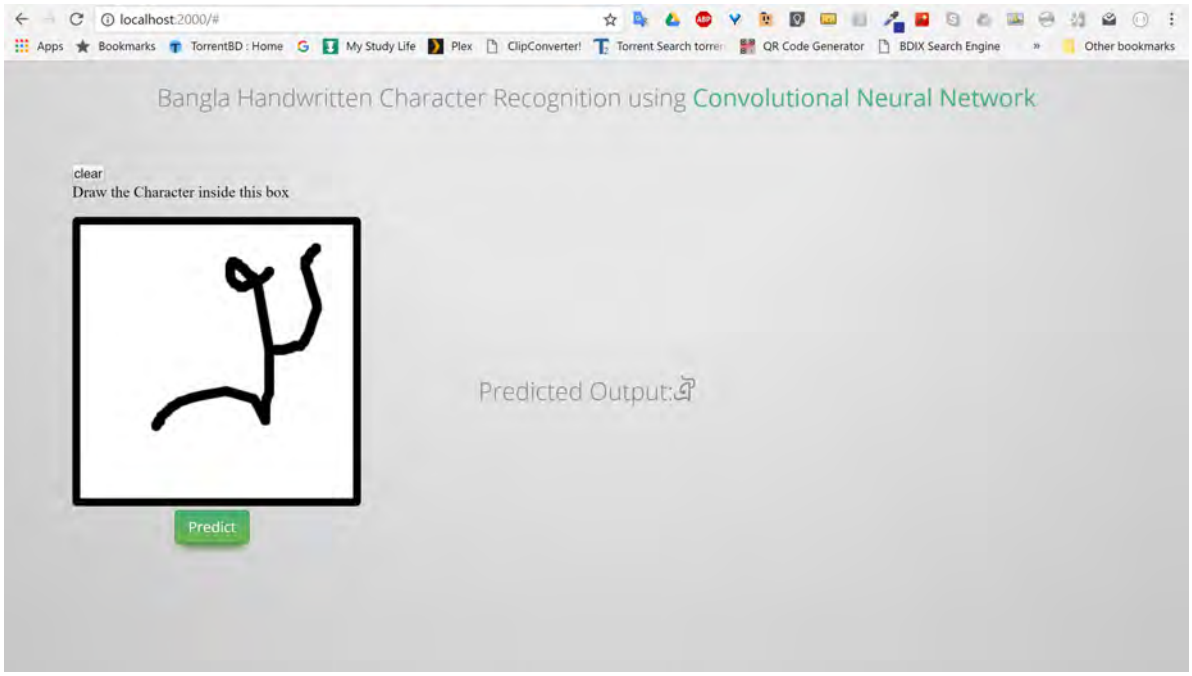

FIGURE 5.1. Prediction of bangla Character

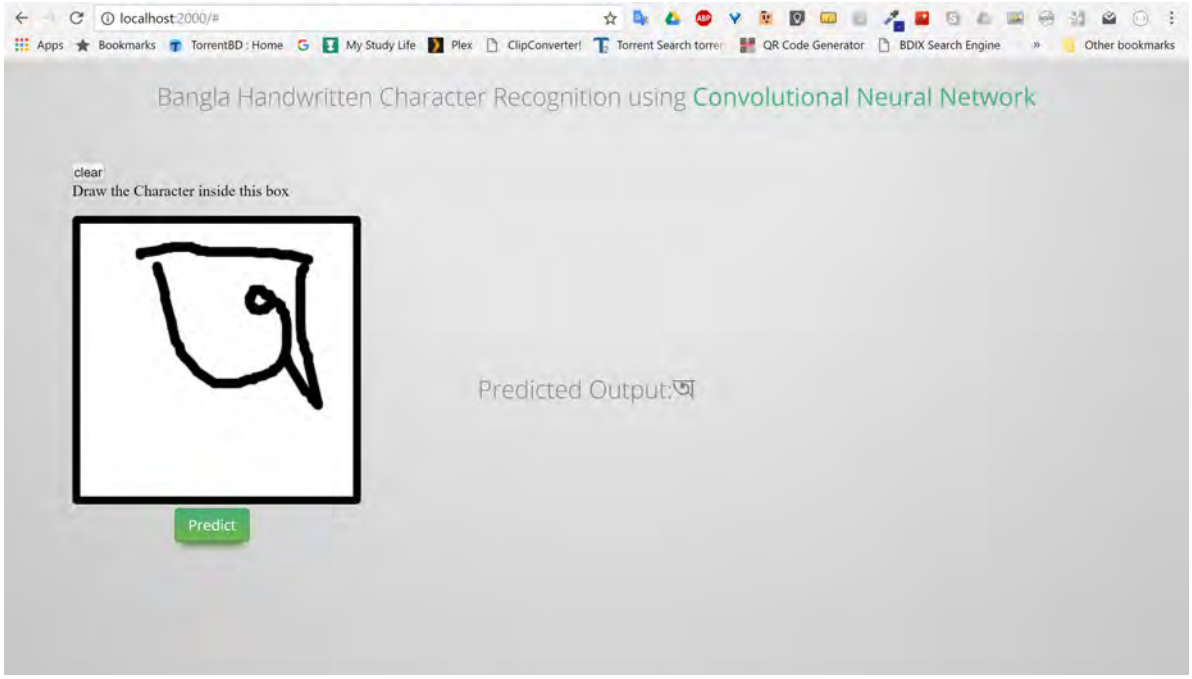

FIGURE 5.2. Prediction of bangla Character

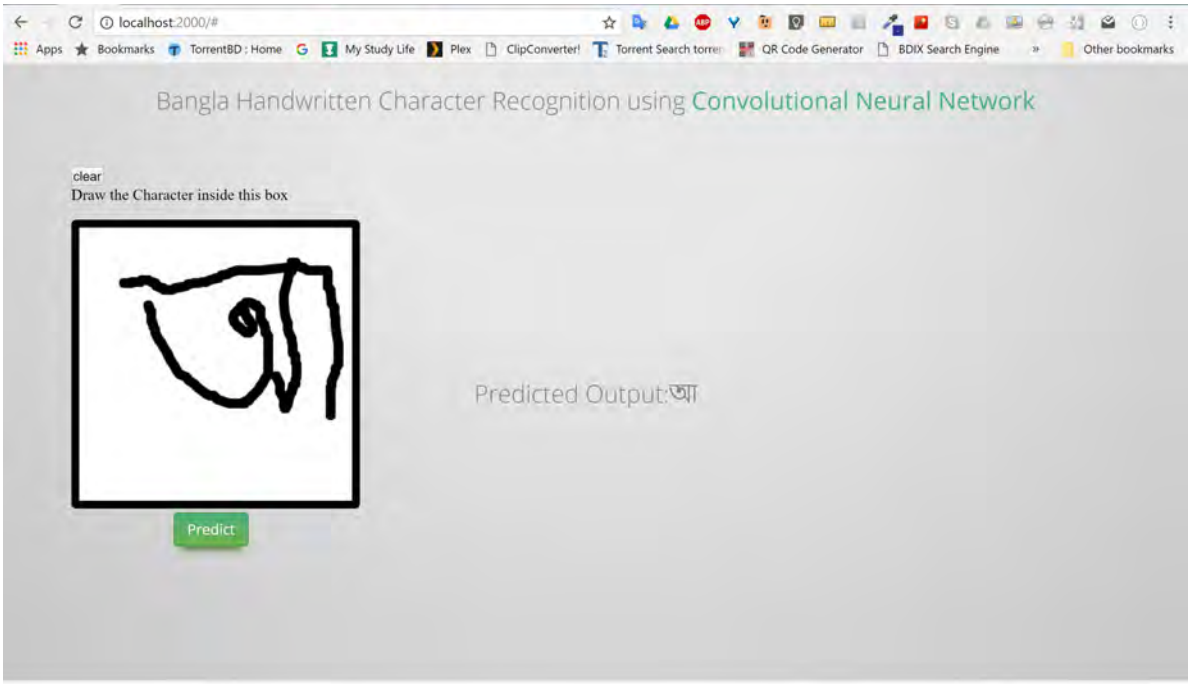

FIGURE 5.3. Prediction of bangla Character

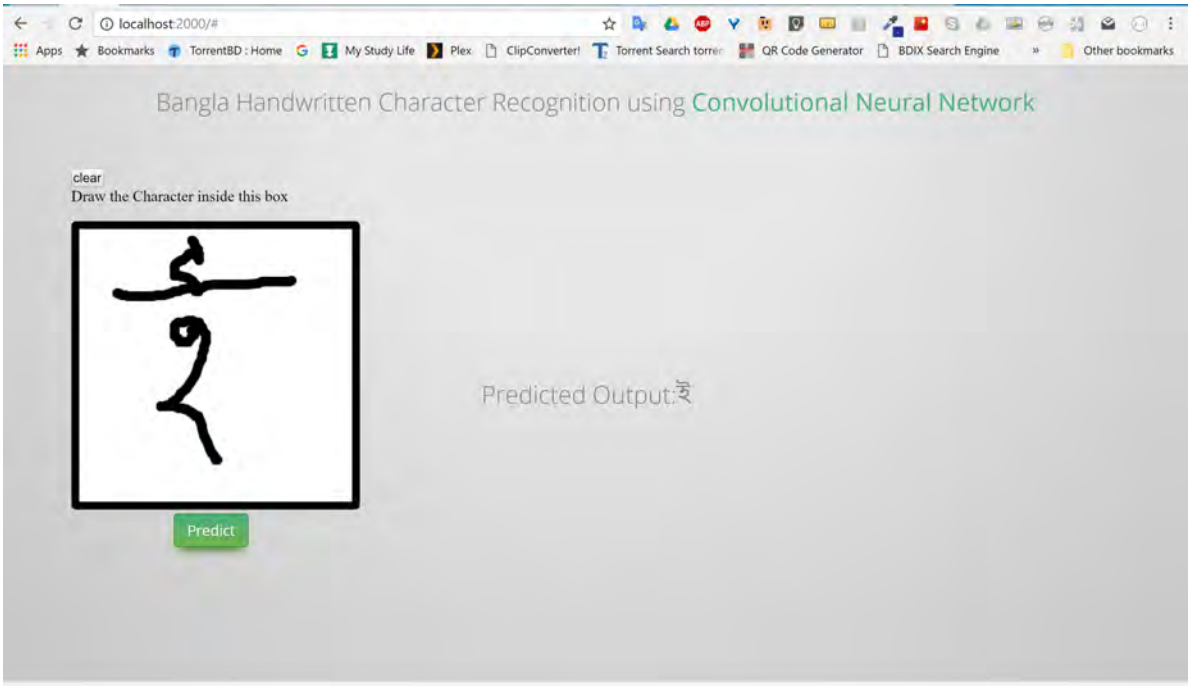

FIGURE 5.4. Prediction of bangla Character

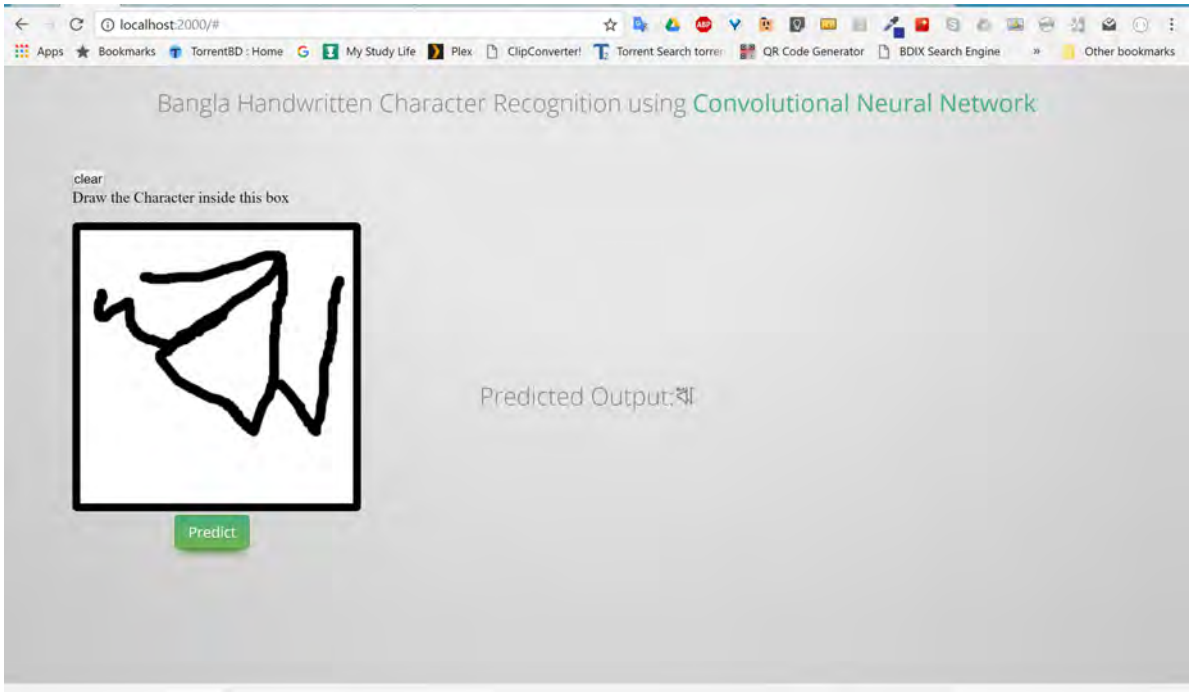

FIGURE 5.5. Prediction of bangla Character

The above demonstration of the predicted character showcases the ability of the model to predict different handwritings for a lot of variations with it. The system was implemented to predict the character even if the drawn image is barely recognizable by the human.This feat is possible mainly of the learning patterns of the convolutional neural network as it can generalize the different classes and patterns from the training dataset.

5.2 Character Localization

Here we are demonstrating the localization of Bangla handwritten and scene text. The MSER algorithm we used detected the edges of each Bangla character and marked the area that will most likely contain the characters.Then the MSER systems send the marked region to our character recognition system to predict each character. And thus the system becomes an End to End Bangla Character detection and recognition system.

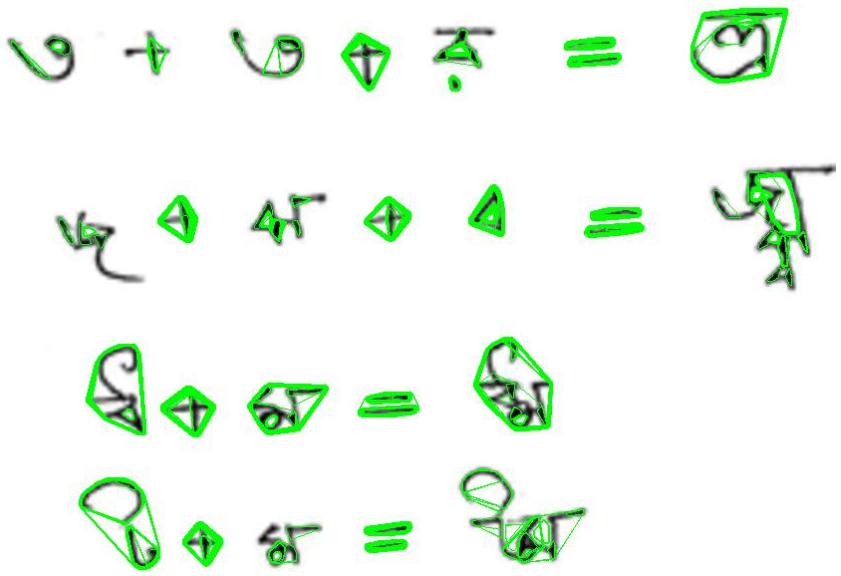

FIGURE 5.6. Bangla Handwritten Character Localization Using MSER

প্রথম আলো

Date:20-08-2013 Page 13, Col 1-4 Size: 2D col*cm

সুশাসনের অভাব ও নিয়ন্ত্রকের দুর্বলতায় ধসেছে রানা প্লাজা

নিজস্ব প্রতিবেদক

অ্যামচেমের ভোজসভায় অধ্যাপক রেহমান সোবহান

সুশাসনের অভাব এবং নিয়ন্ত্রণকারী সংস্থাগুলোর দুর্বলতার বহিঃপ্রকাশ হলো সাভারের রানা প্লাজা ধস। এমনটাই মনে করেন বেসরকারি গবেষণা সংস্থা সেন্টার ফর পলিসি ডায়ালগের (সিপিডি) চেয়ারম্যান অধ্যাপক রেহমান সোবহান।

দেশের এই প্রথিতযশা অর্থনীতিবিদ রানা প্লাজা ধসের জন্য আরও কয়েকটি কারণ চিহ্নিত করেছেন, যার মধ্যে আছে রাজনৈতিক অপসংস্কৃতি, শোচনীয় সামাজিক অবস্থা, জবাবদিহির অভাবসহ বৈশ্বিক অর্থনীতির বর্তমান অবস্থাও। রানা প্লাজা ধসকে দেশের সার্বিক ব্যবস্থার (সিস্টেম) সংকটের লক্ষণ হিসেবেও উল্লেখ করেন তিনি।

আমেরিকান চেম্বার অব কমার্স (আমচেম) ইন বাংলাদেশের মাসিক মধ্যাহ্নভোজসভায় মূল আলোচক হিসেবে রেহমান সোবহান গতকাল এসব কথা বলেছেন। সভার আলোচ্য বিষয় ছিল 'রানা প্লাজার ভবন ধস থেকে শিক্ষা'। রাজধানীর একটি হোটেলে গতকাল সোমবার এ সভা অনুষ্ঠিত হয়।

সুশাসনের ব্যর্থতার প্রসঙ্গে বলতে গিয়ে রেহমান সোবহান বলেন, রানা প্লাজার ভবন গড়ে উঠেছিল অবৈধভাবে। অনুমোদন না নিয়ে তা আরও সম্প্রসারণ করা হয়। সেখানে নিয়ন্ত্রণমূলক পদক্ষেপ ঠিকভাবে নেওয়া হয়নি। পরিদর্শকেরা অনেক কিছুই এড়িয়ে গেছেন। তিনি মনে করেন, পরিদর্শকেরা যদি আরও ভালো বেতন কিংবা প্রণোদনা ভাতা পান, তাহলে তাঁরা আরও ভালোভাবে কারখানা পরিদর্শন করবেন। একই সঙ্গে পরিদর্শনের একটি মানসম্মত গাইডলাইন থাকা প্রয়োজন বলে মনে করেন তিনি।

রাজনৈতিক অপসংস্কৃতির প্রসঙ্গে সিপিডির চেয়ারম্যান বলেন, সোহেল রানার রাজনৈতিক সংশ্লিষ্টতা ছিল। এ কারণেই অল্প সময়ে প্রচুর অর্থের মালিক হয়েছেন তিনি। এটি শুধু একটি মাত্র নয়। এমন নাম আরও অনেক রয়েছে। আর এমনটি হয় সুশাসনের অভাব আর রাজনৈতিক অপসংস্কৃতির কারণে।

রানা প্লাজা ধসের জন্য সমাজের চরিত্রকেও দায়ী করেছেন রেহমান সোবহান। তাঁর মতে, রানা প্লাজায় অবস্থিত কারখানার মালিকেরা মানবিক দিক উপেক্ষা করে কেবল আন্তর্জাতিক ক্রেতাদের কাছ থেকে কার্যাদেশ নেওয়ার দিকেই মনোযোগী ছিলেন। তাই শ্রমিকদের জোর করে কাজ করতে বাধ্য করা হয়েছে। তিনি বলেন, মালিকদের সঙ্গে দর-কষাকষির জন্য সুসংগঠিত ট্রেড ইউনিয়ন থাকলে হয়তো কারখানাগুলোতে জোর করে শ্রমিকদের কাজ করানো যেত না।

রেহমান সোবহান বলেন, শ্রমিকদের শ্রমিক মনে না করে, তাঁদের প্রতিষ্ঠানের অংশীদার মনে করলে অনেক সমস্যারই সমাধান হয়ে যাবে। তাঁদের ন্যূনতমভাবে মুনাফার একটি অংশ দেওয়া যেতে পারে। এসব করলে তাঁদের অন্যে আনুগত্য বাড়বে।

এই অর্থনীতিবিদ হিসাব করে দেখেছেন, বাংলাদেশে তৈরি হওয়া একটি শার্ট বিক্রি করে ওয়াল-মার্ট ৬৩ শতাংশ মুনাফা করে। অথচ, শ্রমিকদের বেতনের পেছনে ব্যয় হয় মাত্র ৫ শতাংশ। এত বিশাল পরিমাণ মুনাফা নিয়ে কেউই কথা বলেন না। তাঁর মতে, এখন সময় এসেছে এ বিষয়টি নিয়ে কথা বলা। তিনি আরও বলেন, গুলশান বাজারসুবিধা, জিএসপি এখন রাজনৈতিক বিষয়ে পরিণত হয়েছে।

সভায় বিকেএমইএ-এর স্বাধক সভাপতি ফজলুল হক বলেন, শুধু ৬৩ শতাংশ নয়, অনেক ক্ষেত্রে বিক্রেতাপ্রতিষ্ঠান ২০০ শতাংশ পর্যন্ত মুনাফা করে। তিনি বলেন, রানা প্লাজার ধসের পর থেকে শুধু ভবনের কাঠামো নিয়ে কথা বলছেন। তাজরীন ফ্যাশনসের পর শুধু আগুনের কথা বলেছিলেন। কিন্তু এমন আরও অনেক বিষয় আছে, যেগুলো চিহ্নিত করে ব্যবস্থা নেওয়া প্রয়োজন।

সভায় বিশিষ্ট আইনজীবী ড. কামাল হোসেন, আমচেমের সভাপতি আফতাব উল ইসলাম, নবাহী পরিচালক এ ফয়ুর রেমুখ বক্তব্য দেন।

FIGURE 5.7. Bangla scene Character Localization Using MSER

CHAPTER 6

CONCLUSION

Convolutional neural network has powerful and robust ability to recognize visual pattern from pixel image. In this paper we have considered a multilayer convolutional neural network without any feature selection for Bangla handwritten character classification. The proposed method gives competitive performance with the existing other methods in terms of test accuracy. And it also tested on a large dataset. Adding more convolutional layer and hyper parameter tuning may give improved performance in future. And as machine learning networks can be used as a general purpose learning model, we also want to conclude that the same multilayer convolutional network can be used for Bangla scene-text detection and recognition with state of the art performance.

BIBLIOGRAPHY

[1] H. CHEN, S. S. TSAI, G. SCHROTH, D. M. CHEN, R. GRZESZCZUK, AND B. GIROD, *Robust text detection in natural images with edge-enhanced maximally stable extremal regions*, in Image Processing (ICIP), 2011 18th IEEE International Conference on, IEEE, 2011, pp. 2609–2612.

[2] X. CHEN AND A. L. YUILLE, *Detecting and reading text in natural scenes*, in Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, vol. 2, IEEE, 2004, pp. II–II.

[3] T. DE CAMPOS, B. R. BABU, AND M. VARMA, *Character recognition in natural images*, (2009).

[4] T. HE, W. HUANG, Y. QIAO, AND J. YAO, *Text-attentional convolutional neural network for scene text detection*, IEEE transactions on image processing, 25 (2016), pp. 2529–2541.

[5] W. HUANG, Z. LIN, J. YANG, AND J. WANG, *Text localization in natural images using stroke feature transform and text covariance descriptors*, in Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1241–1248.

[6] W. HUANG, Y. QIAO, AND X. TANG, *Robust scene text detection with convolution neural network induced mser trees*, (2014), pp. 497–511.

[7] M. JADERBERG, K. SIMONYAN, A. VEDALDI, AND A. ZISSERMAN, *Reading text in the wild with convolutional neural networks*, International Journal of Computer Vision, 116 (2016), pp. 1–20.

[8] L. KANG, Y. LI, AND D. DOERMANN, *Orientation robust text line detection in natural images*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 4034–4041.

[9] K. I. KIM, K. JUNG, AND J. H. KIM, *Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25 (2003), pp. 1631–1639.

[10] Y. LI, W. JIA, C. SHEN, AND A. VAN DEN HENGEL, *Characterness: An indicator of text in the wild*, IEEE transactions on image processing, 23 (2014), pp. 1666–1677.

[11] J. MATAS, O. CHUM, M. URBAN, AND T. PAJDLA, *Robust wide-baseline stereo from maximally stable extremal regions*, Image and vision computing, 22 (2004), pp. 761–767.

[12] A. MISHRA, K. ALAHARI, AND C. JAWAHAR, *Scene text recognition using higher order language priors*, in BMVC 2012-23rd British Machine Vision Conference, BMVA, 2012.

[13] D. NISTÉR AND H. STEWÉNIUS, *Linear time maximally stable extremal regions*, Computer Vision–ECCV 2008, (2008), pp. 183–196.

[14] M. OZUYSAL, P. FUA, AND V. LEPETIT, *Fast keypoint recognition in ten lines of code*, in Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, Ieee, 2007, pp. 1–8.

[15] Y.-F. PAN, X. HOU, AND C.-L. LIU, *A hybrid approach to detect and localize texts in natural scene images*, IEEE Transactions on Image Processing, 20 (2011), pp. 800–813.

[16] J. SOCHMAN AND J. MATAS, *Waldboost-learning for time constrained sequential detection*, in Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 2, IEEE, 2005, pp. 150–156.

[17] L. SUN, Q. HUO, W. JIA, AND K. CHEN, *A robust approach for text detection from natural scene images*, Pattern Recognition, 48 (2015), pp. 2906–2920.

[18] C. SZEGEDY, W. LIU, Y. JIA, P. SERMANET, S. REED, D. ANGUELOV, D. ERHAN, V. VANHOUCKE, AND A. RABINOVICH, *Going deeper with convolutions*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9.

[19] K. WANG, B. BABENKO, AND S. BELONGIE, *End-to-end scene text recognition*, in Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 1457–1464.

[20] T. WANG, D. J. WU, A. COATES, AND A. Y. NG, *End-to-end text recognition with convolutional neural networks*, (2012), pp. 3304–3308.

[21] C. YI AND Y. TIAN, *Text string detection from natural scenes by structure-based partition and grouping*, IEEE Transactions on Image Processing, 20 (2011), pp. 2594–2605.

[22] X.-C. YIN, X. YIN, K. HUANG, AND H.-W. HAO, *Robust text detection in natural scene images*, IEEE transactions on pattern analysis and machine intelligence, 36 (2014), pp. 970–983.

[23] Z. ZHANG, W. SHEN, C. YAO, AND X. BAI, *Symmetry-based text line detection in natural scenes*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2558–2567.