Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 167 (2020) 2403–2409

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2019)

Handwritten Character Recognition from Images using CNN-ECOC

Mayur Bhargab Bora, Dinthisrang Daimary, Khwairakpam Amitab*, Debdatta Kandar

Department of Information Technology, North-Eastern Hill University, Shillong-793022, India

Abstract

Recently, deep learning and character recognition have drawn the attention of many researchers. The deep neural networks have state-of-the-art performance in solving many classification and recognition problems. The Optical Character Recognition (OCR) takes an optical image of character as input and produces the corresponding character as output. It has a wide range of applications including traffic surveillance, robotics, digitization of printed articles, etc. The OCR can be implemented by using Convolutional Neural Network (CNN), which is a popular deep neural network architecture. The traditional CNN classifiers are capable of learning the important 2D features present in the images and classify them, the classification is performed by using soft-max layer. In this article, we have presented OCR by combining CNN and Error Correcting Output Code (ECOC) classifier. The CNN is used for feature extraction and the ECOC is used for classification. In order to find suitable CNN for extracting features, which can be used in combination with ECOC classifier for recognition of handwritten characters accurately, several popular CNN classifiers have been explored. The CNN-ECOC are trained and validated by using NIST handwritten character image dataset. The simulation result shows that CNN-ECOC gives higher accuracy as compared to the traditional CNN classifier.

Keywords: Character recognition; Classification; CNN; Deep learning; ECOC; OCR; SVM;

1. Introduction

The OCR is a process of classifying the optical patterns present in a digital image to the corresponding characters.The character recognition is achieved through important steps of feature extraction and classification [1]. OCR system simulates the human capability to recognize printed forms of text and it has become one of the most successful applications in the field of object recognition. Applications of OCR include identifying the vehicle registration number from the image of number plate which helps in controlling traffic [2], converting printed academic records into text for storing in an electronic database, decoding ancient scripture, automatic data entry by optical scanning of cards, bank checks, etc. The OCR systems avoid typing mistakes and save time. In order to implement OCR, neural

* Corresponding author. Tel.: +91-364-272-3628.
E-mail address: kamitab@nehu.ac.in

1877-0509
10.1016/j.procs.2020.03.293

networks can be used [3].
Handwritten character recognition is challenging and researchers have been exploring different techniques in the pass few decades. Recently, deep neural networks have drawn the attention of many researchers due to their capability of solving computer vision problem such as object detection, classification, recognition [4], etc. undoubtedly well. CNN is one of the most popular type of deep neural network, it can learn and extract features from the 2D images.The CNN classifier can effectively recognize characters present in the image. The architecture of traditional CNN classifier consists of convolutional layers for extracting features and fully connected layers followed by a soft-max layer for classification. CNN is an effcient feature extractor [5].

In this article CNN-ECOC have been presented, which is a hybridization of CNN architecture and ECOC classifier. The CNN is used for feature extraction and ECOC is used for classification. The soft-max layer in traditional CNN is replaced by the ECOC classifier in CNN-ECOC. ECOC basically converts multi-class classification problem into a binary classification problem with the help of various coding schemes accompanied by a linear learner like Support Vector Machine (SVM). The SVM maps the inputs to high dimensional space where the differences between the classes can be revealed. The SVM can avoid over fitting automatically and has a high prediction accuracy [6]. The SVM also has better generalization capability than neural network [7].
The rest of the paper is organized as follows: Section 2 gives an introduction about the different layers of CNN. The proposed combination of CNN and ECOC is presented in section 3. Section 4 presents the implementation and result. This paper is concluded in section 5.

2. Convolutional Neural Networks

CNN are made up of a large number of interconnected neurons that have learnable weights and biases. In the architecture of CNN the neurons are organized as layers. It consist of an input layer, many hidden layers and an output layer. If the network has a large number of hidden layers the same are generally referred as deep neural networks. The neurons in the hidden layers of CNN are connected to a small region (receptive field) of the input space generated from the previous layer instead of connecting to all, as in the fully connected network like Multi Layered Perceptron (MLP) networks. This approach reduces the number of connection weights (parameters) in CNN compared to MLP. As a consequence, CNN takes less time to train for the networks of similar size [8]. The input to the typical CNN are two dimensions (2D) array of data such as images. Unlike the regular neural network the layers of a CNN are arranged in three dimensions (width, height and depth).

The followings are the type of layers which as commonly found in the CNNs.

- Input Layer is basically a buffer to hold the input and pass it on to the next layer.
- Convolution Layer performs the core operation of feature extraction. It performs Convolution operation of the input data. The convolution operation is executed by sliding the kernel over the input, and performs sum of the product at every location. The step size with which the kernel slides are known as stride. Numerous convolution operations are performed on the input by using different kernel, which results in different feature maps, the number of feature map produced in a convolutional layer is also known as the depth of the layer.
- Rectified Linear Unit (ReLU) is an activation function used to introduce non linearity. It replaces negative value with zero, which can speed up the learning process. The output of every convolution layer is passed through the activation function.
- Pooling layer reduces the spatial size of each feature map, which in turn reduces the computation in the network. Pooling also uses a sliding window that moves in stride across the feature map and transform it into representative values. Min pooling, average pooling and max pooling are commonly used.
- Fully connected layer connects every neuron in the layer to all the neurons in the previous layer. It learns the non-linear combination of the features and used for classifying or predicting the output. For classification problems, the fully connected layer is generally followed by a soft-max layer, it produces the probability of each class for the given input. And for regression problems, it is followed by a regression layer to predict the output.

The architecture of the CNN is organized according to the problem to be addressed. Like any other neural networks the CNN is intelligent and they can learn. Learning is achieved through training (supervised). The CNN are feedforward networks and uses back-propagation training algorithm. The training is performed in two passes; forward and backward pass. In the forward pass the network weight and bias are initialized with small random numbers and compute the network output by using training input. The error is computed by comparing the network output with the desired training output. In the backward pass the error propagates backward and all the weights and bias are adjusted to minimize the error. The process is repeated until the desired result is obtained. Once the network is trained with a suitable dataset, it can be used for solving a specific problem.

Many researches have developed different CNN architectures for classification. Some of the most popular deep CNN networks include AlexNet [8], ZfNet [9] and LeNet [10].

- **AlexNet**
  AlexNet accepts input of size $227 \times 227 \times 3$ and consist of 25 layers. It has 5 convolution layers which are followed by ReLU layer and max pooling layer. The first two convolution layers are also followed by cross-channel normalization layer. Cross channel normalization carries out channel wise normalization. It replaces each element with a normalized value obtained from neighbouring cells. It also has three fully connected layers followed by ReLU Layer and the first two fully connected layer is also followed by dropout layers. The output of the last fully connected layer is given as an input to softmax which produces the probability distribution of 1000 classes. The AlexNet was designed for recognizing objects in ImageNet [11].
- **ZfNet**
  ZfNet has similar architecture to AlexNet but differs in the filter size and the number of filters used. In the first convolution layer, the AlexNet uses filter of size 11×11 with a stride of 4 and the ZfNet uses 7×7 and stride of 2. The ZfNet uses 512,1024 and 512 filters in the third,fourth and fifth convolution layer whereas the AlexNet uses 384,384 and 256 filters respectively. The ZfNet was also designed for recognizing objects in ImageNet [11].
- **LeNet**
  LeNet accepts a grayscale image of size 32×32 as an input. The architecture consists of two sets of convolutional and average pooling layers, followed by a flattening convolutional layer, then follows two fully connected layers and a softmax layer for classification. The LeNet is a pioneering work in recognizing hand written digits from the images by using CNN.

Many researches have proposed different CNN architecture for recognizing handwritten characters and have achieve high accuracy. Yuan et al. in 2012 [12], modified the LeNet architecture by incorporating error correcting codes in the output of the CNN, thus the CNN has the ability to reject incorrectly recognized results. They have trained and evaluated on UNIPEN [11] lowercase and uppercase datasets, with recognition rates of 93.7% for uppercase and 90.2% for lowercase, respectively. Researchers have also used CNN for recognizing indigenous languages, one of such work is proposed by Rahman et al. in 2015 [13], the CNN model consist of two convolutional layer followed by two sub sampling layers. Training and test dataset consisting of 17500 and 2500 images containing handwritten Bangla text, and achieved an accuracy rate of 85.36% in recognizing over 50 different Bangla characters. Deng [14][15] carried out recognition of English letters, digits, and special characters by combining CNN and ECOC. The experimental result shows, by using ECOC the reliability and accuracy can be improved.

3. Combining CNN with ECOC

The ECOC considers the problem of classification as a communication problem, ECOC addresses the multi-class problem by dividing into several binary problems. ECOC can be defined as a bit string of Length L which is the codeword, where L is associated with each class X_i with i=(1,...n) so that every class is represented by a different codeword. If there are n original classes, then, the set of codewords is arranged in an $n \times L$ coding matrix C = $\{c_{a,b}\}$, where $c_{a,b} \epsilon \{-1, +1\}$. The column of C represents L binary problems,each requiring a specific dichotomizer.

In the proposed CNN-ECOC, the codewords are designed with maximum minimum code distances, which means that the codewords are separated by a fixed hamming distance from each other [16]. The output codeword generated

by the classifier is compared to the coding matrix shown in Table: 1, and the corresponding class with the nearest codeword is the predicted class. To save space we have shown the codeword of the first fourteen classes only.

Table 1: Codeword of First Fourteen Classes Generated by The ECOC classifier

Classes	Codewords
A	1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
B	-1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
C	-1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
D	-1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
E	-1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
F	-1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
G	-1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
H	-1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
I	-1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
J	-1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
K	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
L	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
M	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
N	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1

In the proposed approach the ECOC classifier is trained with the extracted features from the CNN. Given any input image the ECOC classifier works by extracting the features of the input image and then feeding those to all the binary learners. The binary learners are trained using linear SVM, where one class is taken as positive and others are taken as negative and are separated using a hyper-plane. Once the binary learner is trained, each learner produces a probability. All the probabilities are collected into a string and converted into a codeword by using a suitable threshold. The codeword generated by the ECOC classifier is compared with the coding matrix shown in Table: 1 and the class corresponding to the nearest codeword is the predicted output.

The steps involved in recognition of handwritten character is shown in Figure 1. The input image is fed to the CNN, which extract the features, the ECOC performs the recognition by using the features and the corresponding character is produced as an output.

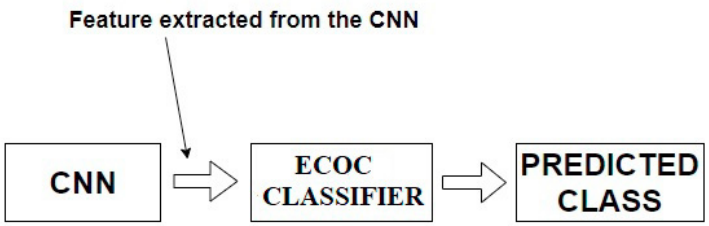

Fig. 1: CNN-ECOC Classifier

4. Implementation and Result

Four different CNN architecture have been implemented for extracting features and a ECOC classifier. The features are collected from the output of the last fully connected layer in the CNN. NIST handwritten character dataset [17] has been used for training the CNN and ECOC.Ciresan [7] in 2011 and Fanany in 2017 [18] had also proposed handwritten character recognition by combining CNN and ECOC. Training and testing on NIST dataset was done and accuracy of

88% and 93% was achieved, respectively.

The aim of this work is to increase the accuracy of CNN character recognition system by using ECOC classifier, the dataset is divided into 26 folders, each containing 2473 (1483 training images and 990 testing images) different handwritten character images of an upper case English alphabet. NVIDIA GeForce GT 730 graphics driver is used to train the network. All the experiment have been carried out on single GPU with 8 GB of RAM. The CNNs are first trained to learn the features and the extracted features are used to train the ECOC classifier. Firstly, the dataset is prepossessed by performing operation like resizing the images and converting the RGB images into grayscale images as required by the CNN networks. Several popular CNN models have been explored which can be used as a feature extractor for using along with ECOC classifier. The simulation parameters and the details of the training are as follows

- **LeNet of Type 1 with ECOC Approach**
  The LeNet of type one, takes a grays scale image of size 32×32 as input. It was trained with gradient descent back propagation algorithm, the mini-batch size and max epoch is set to 64 and 20 respectively. A training accuracy rate of 74.63% and testing accuracy of 73.78% was obtained. Then the features are extracted from the last fully connected layer of the trained LeNet and fed to a ECOC classifier. The ECOC is trained with Liner SVM learner and uses one vs all coding method and got a training accuracy rate of 67.43% and testing accuracy of 67.43%. It was observed that accuracy has decreased after replacing softmax with ECOC, so the LeNet architecture has been modified as discussed in LeNet Type 2. The LeNet takes 26 minutes to train and the extracted features are used to train the ECOC classifier, which takes additional 20 seconds.
- **LeNet of Type 2 with ECOC Approach**
  The LeNet of type two, takes an RGB image of size $128 \times 128 \times 3$ as input. A ReLU layer has been added followed by a dropout layer with a dropout rate of 50% after the first fully connected layer. The network is trained for 2 hours 60 minutes with gradient descent back propagation algorithm. The mini-batch size and max epoch are set to 64 and 20 respectively and a training accuracy rate of 89.00% and testing accuracy of 85.86% have been obtained. Then the features is extracted from the last fully connected layer of the trained network and fed to an ECOC classifier. The ECOC is trained for 20 seconds with liner SVM learner and uses one vs all coding method and a training accuracy rate of 85.88% and testing accuracy of 85.88% have been achieved. It is observed that the accuracy is increased in the modified LeNet architecture of type 2 by 0.02%.
- **AlexNet with ECOC Approach**
  The AlexNet, takes an RGB image of size $128 \times 128 \times 3$ as input. The network has been trained with gradient descent back propagation algorithm, the mini-batch size and max epoch is set to 64 and 20 respectively for 4 hours 20 minutes. A training accuracy rate of 98.97% and testing accuracy of 97.06% have been achieved. Then the features are extracted from the last fully connected layer of the trained AlexNet and fed to an ECOC classifier. The ECOC is trained for 25 seconds with liner SVM learner with one vs all coding method and a training accuracy rate of 97.71% and testing accuracy of 97.71% is achieved. It is observed that the testing accuracy has increased by 0.6%.
- **ZfNet with ECOC approach**
  The ZFnet takes an RGB image of size $128 \times 128 \times 3$ as input. The network is trained with gradient descent back propagation algorithm for 4 hours 40 minutes, by setting the mini-batch size and max epoch to 64 and 20 respectively. Training accuracy rate of 99.65% and testing accuracy of 97.65% is achieved. The extracted features from the last fully connected layer of the trained ZfNet are fed to an ECOC classifier. The ECOC is trained for 28 seconds with liner SVM learner using one vs all coding method and got a training accuracy rate of 97.71% and testing accuracy of 97.71%. It is observed that the testing accuracy is increased by 0.06%.

The summary of the simulation result obtained from the different networks that have been implemented are shown in Table:2. From the simulation result presented in Table:2 it has been observed that in most of the networks, the classification accuracy can be improved by replacing the softmax layer with ECOC. The ECOC classifier gives the highest testing accuracy if the features are extracted by using AlexNet architecture.

Table 2: Accuracy Result of various Architectures

Name of the Architecture	Training Accuracy	Testing accuracy
LeNet of Type 1	74.63%	73.78%
LeNet-ECOC of Type 1	67.43%	67.43%
LeNet of Type 2	89.00%	85.86%
LeNet-ECOC of Type 2	85.88%	85.88%
AlexNet	98.97%	97.06%
AlexNet-ECOC	97.71%	97.71%
ZfNet	99.65%	97.65%
ZfNet-ECOC	97.71%	97.71%

5. Conclusion

In this article handwritten character recognition systems have been presented by using CNN-ECOC, which is a combination of CNN and ECOC classifier. The CNN is used for feature extraction and ECOC for recognition of characters. In order to find a suitable feature extractor, three popular CNN architectures have been explored, namely LeNet, AlexNet and ZfNet. From the simulation result, it has been observed that LeNet gives a low accuracy rate. Hence it has been modified by adding dropout layer and ReLu layer after the first fully connected layer, and it has resulted in higher accuracy rate. It has also been observed that the accuracy of ECOC classifiers is higher as compared to the CNN softmax classifier. Amongst the implemented network the AlexNet is the most suitable CNN for combining with ECOC, in order to recognize handwritten characters.

References

[1] Chaudhuri, Arindam and Mandaviya, Krupa and Badelia, Pratixa and Ghosh, Soumya K and others. (2017) "Optical Character Recognition System. *In Optical Character Recognition Systems for Different Languages with Soft Computing*Springer: 941.

[2] Li, Haixiang and Yang, Ran and Chen, Xiaohui. (2017) "License plate detection using convolutional neural network. *3*rd IEEE International Conference on Computer and Communications (ICCC),IEEE:17361740.

[3] Rajavelu, A and Musavi, Mohamad T and Shirvaikar, Mukul Vassant. (1989) " A neural network approach to character recognition.Neural Network 5,Elsevier (2): 387393.

[4] Bai,Jinfeng and Chen, Zhineng and Feng, Bailan and Xu, Bo.(2014) "Image character recognition using deep convolutional neural network learned from different languages. *I*EEE International Conference on Image Processing (ICIP):25602564.

[5] Maitra, Durjoy Sen and Bhattacharya, Ujjwal and Parui, Swapan K. (2015) "CNN based common approach to handwritten character recognition of multiple scripts.*I*3th International Conference on Document Analysis and Recognition (ICDAR),IEEE:10211025.

[6] Jakkula,Vikramaditya. (2006)"Tutorial on support vector machine (svm). *S*chool of EECS, Washington State University **37**.

[7] Ciresan,Dan Claudiu and Meier,Ueli and Gambardella,Luca Maria and Schmidhuber,Jurgen. (2011)"Convolutional neural network committees for handwritten character classification. *I*nternational Conference on Document Analysis and RecognitionIEEE :11351139.

[8] Krizhevsky, Alex and Sutskever, Ilya and Hinton, Geoffrey E. (2012) "Imagenet classification with deep convolutional neural networks.*A*dvances in neural information processing systems:10971105.

[9] Zeiler, Matthew D and Fergus,Rob. (2014) "Visualizing and understanding convolutional networks. *E*uropean conference on computerSpringer, vision:818833.

[10] LeCun, Yann and Bottou, Leon and Bengio, Yoshua and Haffner, Patrick and others. (1998)"Gradient-based learning applied to document recognition. *P*roceedings of the IEEE,Taipei, Taiwan, **86** (11): 22782324.

[11] Guyon, Isabelle and Schomaker, Lambert and Plamondon, Rejean and Liberman, Mark and Janet, Stan. (1994) "UNIPEN project of on-line data exchange and recognizer benchmarks." *P*roceedings of the 12th IAPR International Conference on Pattern RecognitionIEEE,(2):2933.

[12] Yuan, Aiquan and Bai, Gang and Jiao, Lijing and Liu, Yajie. (2012) "Onine handwritten English character recognition based on convolutional neural network. *1*0th IAPR International Workshop on Document Analysis SystemsIEEE: 125129.

[13] Rahman, Md Mahbubar and Akhand, MAH and Islam, Shahidul and Shill, Pintu Chandra and Rahman, MH and others. (2015) "Bangla handwritten character recognition using convolutional neural network. *I*nternational Journal of Image, Graphics and Signal Processing(IJIGSP),**7** (8): 4249.

[14] Deng, Huiqun and Stathopoulos, George and Suen, Ching Y. (2009) "Errorcorrecting output coding for the convolutional neural network for optical character recognition. *1*0th International Conference on Document Analysis and Recognition,IEEE: 581585.

[15] Deng, Huiqun and Stathopoulos, George and Suen, Ching Y. (2010) "Applying error-correcting output coding to enhance convolutional neural network for target detection and pattern recognition.20th International Conference on Pattern Recognition,IEEE :42914294.

[16] Dietterich, Thomas G and Bakiri, Ghulum. (1994) "Solving multiclass learning problems via error-correcting output codes. *J*ournal of artificial intelligence research (2):263286.

[17] Grother,Parick J and Hanaoka,Kayee K. (2016) "NIST special database 19 handprinted forms and characters database. *N*ational Institute of Standards and Technology.

[18] Fanany,Mohamad Ivan and others. (2017) "Handwriting recognition on form document using convolutional neural network and support vector machines (CNN-SVM). *5*th International Conference on Information and Communication Technology (ICoIC7),IEEE:16.