International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 5, Issue 11, ISSN No. 2455-2143, Pages 57-61
Published Online March 2021 in IJEAST (http://www.ijeast.com)

YORÙBÁNET: A DEEP CONVOLUTIONAL NEURAL NETWORK DESIGN FOR YORÙBÁ ALPHABETS RECOGNITION

Oyeniran, Oluwashina Akinloye
Department Of Computer Science, Ajayi Crowther University, Oyo Town, Oyo State, Nigeria

Oyebode, Ebenezer Olukunle
Department Of Computer Science, Ajayi Crowther University, Oyo Town, Oyo State, Nigeria

Abstract— **Numerous works have been proposed and implemented in computerization of various human languages, nevertheless, miniscule effort have also been made so as to put Yorùbá Handwritten Character on the map of Optical Character Recognition. This study presents a novel technique in the development of Yorùbá alphabets recognition system through the use of deep learning. The developed model was implemented on Matlab R2018a environment using the developed framework where 10,500 samples of dataset were for training and 2100 samples were used for testing. The training of the developed model was conducted using 30 Epoch, at 164 iteration per epoch while the total iteration is 4920 iterations. Also, the training period was estimated to 11296 minutes 41 seconds. The model yielded the network accuracy of 100% while the accuracy of the test set is 97.97%, with F1 score of 0.9800, Precision of 0.9803 and Recall value of 0.9797.**

Keywords— **Yorùbá, Alphabets, Deep Learning, Yorùbánet**

I. INTRODUCTION

Yorùbá language is a member of the Volta-Niger branch of the Niger-Congo family of languages. It is spoken by about 28 million people in southwest Nigeria, Benin, Togo, the UK, Brazil, and the USA [1]. The tonality of Yorùbá alphabets makes its recognition more difficult than that of the printed English alphabet [2]. Despite its popularity, little was done by scholars and researchers in the recognition of Yorùbá Handwritten Character. Numerous handwritten character recognition systems had been developed to recognize characters or texts of some languages such as Chinese, French, English, Arabic, Japanese, Chinese, Korean, Sinhala, and so on, which have yielded outstanding results. Nevertheless, it was noted that there are just a few character recognition systems for Yorùbá Language [3].

In an attempt to put Yorùbá language alphabets on the map of natural language processing and optical recognition, [4] presented Bayesian and Decision Tree techniques, with a recognition rate of 94.44% for the developed recognition system for handwritten characters of Yorùbá upper case letters. [5] through the use of Geometric and Statistical Features, extracted the global and local features of Yorùbá language alphabets and topological features with tolerance to variation style and distortion and thus achieved the recognition rate of 96%.

[6] adopted Correlation and Template Matching Techniques to develop an Optical Character Recognition for Yorùbá texts and convert English numerals in the document to Yorùbá numerals. An unspecified very high accuracy was reported when subjected to test on various sizes of Yorùbá alphabets, and numerals. [3] experimented 600 handwritten images for Yorùbá alphabets, where 480 samples were used for training and 120 samples were used for testing using Support Vector Machine, with 76.7% recognition rate, and a rejection rate of 23.3%. [2] developed an offline Yorùbá character recognition system with a recognition accuracy of 87.7%, using Freeman chain code and K-Nearest Neighbor.

[7] explore image grayscale, binarization, de-skew, and segmentation. Thus, Optical Character Recognition enables the system to read the images and convert them to text data. The model was evaluated using the information retrieval metrics: Precision and Recall. Results showed a significant performance with a recall of 100% in the sample document used, and precision results that vary between 76%, 97%, and 100% in the sample document.

[8] presented an offline Yorùbá Word Recognition System using SVM. The database comprises six hundred (600) handwritten images with different variations and styles which were obtained from several

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 5, Issue 11, ISSN No. 2455-2143, Pages 57-61
Published Online March 2021 in IJEAST (http://www.ijeast.com)

individuals. The words and characters were pre-processed and the geometric features were extracted using Zoning and Gradient Feature Extraction. The feature vectors were used to train the SVM and the words were fed into the SVM classifier for recognition. The system was evaluated based on recognition accuracy. The recognition rate ranges between 66.7%, 83.3%, 85.7%, 87.5% and 100% when tested with twenty (20) handwritten words.

[9] developed Compass Edge Detection Algorithm to enhance the recognition rate of Yorùbá characters. The algorithm developed achieved 0.923 edge detection error rate. The level of accuracy will have been better if noise removal or reduction steps were included in the algorithm.

[10] acquired image dataset at 300 dots per inches (dpi) by generating image textline from Yorùbá New Testament Bible (Bibeli Mimo) corpus using Unicode UTF8. The Long Short Term Memory (LSTM) model, a variant of Recurrent Neural Network (RNN) was used to formulate the Standard Yorùbá character image recognition model. The results show that the Character Error Rate (CER) of 3.138% for the font Times New Roman which gives better recognition than the other font style metric performance. The model achieved an Optical Character Recognition result of (7.435% CER) DejaVuSans font style image dataset, while for Ariel font image dataset, a result of 15.141% was achieved. The Times New Roman font recorded an error rate of 1.182%, the DejaVuSans font style at an error rate of 4.098% while the Ariel font at 5.87%.

The preceding authors have done very well in the development of the Yorùbá Language recognition system. Nevertheless, there is a need to develop a state of the heart recognition system for Yorùbá Language Alphabets so as to standardized the previous developments. Thus, this work aims to develop a Deep Learning based Offline Yorùbá Handwritten Alphabets Recognition System so as to fill all the gaps in the existing knowledge.

II. METHODOLOGY

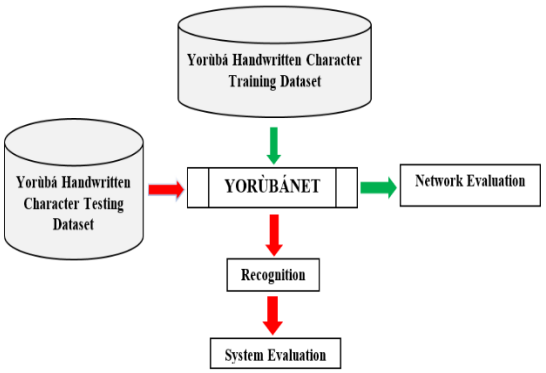

Figure 1: Framework for the development of CNN based Yorùbá Alphabet Recognition System

A. Data Acquisition

The acquired datasets are 12,600 samples. Where 80% (10,500 samples) of the datasets were reserved for training while the remaining 20% (2,100 samples) were separated for further validation. Since there are 70 classes in the dataset, it implies that for the training and evaluation there are 150 samples in each class where there are 30 samples in each class for the verification dataset. The dataset is available on https://github.com/Oluwashina90/YORÙBÁ-HANDWRITTEN-CHARACTER-DATABASE.

B. Network Architecture

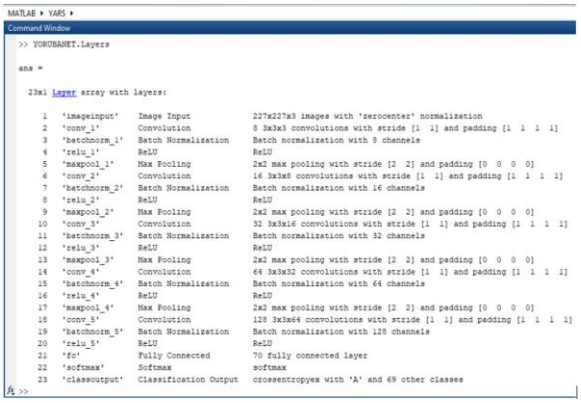

Figure 2: YORÙBÁNET Architecture

Input Layer: The input layer is tagged *imageInputLayer*. The image input layer defines the size of the input images of a convolutional neural network and contains the raw pixel values of the images. For the purpose of this study the size of the input is 227 by 227 by 3.

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 5, Issue 11, ISSN No. 2455-2143, Pages 57-61
Published Online March 2021 in IJEAST (http://www.ijeast.com)

Convolution Layer: This layer is tagged *convolution2dLayer*. A convolutional layer consists of neurons that connect to subregions of the input images or the outputs of the layer before it. A convolutional layer learns the features localized by these regions while scanning through an image.

Batch Normalization Layer: The layer is tagged *batchNormalizationLayer*: This layer speed up network training and reduce the sensitivity to network initialization. The layer first normalizes the activations of each channel by subtracting the mini-batch mean and dividing by the mini-batch standard deviation. Then, the layer shifts the input by an offset β and scales it by a scale factor γ. β and γ are themselves learnable parameters that are updated during network training.

Rectified Linear Unit Layer: The layer is tagged *reluLayer* Convolutional and batch normalization layers are usually followed by a nonlinear activation function such as a rectified linear unit (ReLU), specified by a ReLU layer. The ReLU layer does not change the size of its input but performs a threshold operation to each element, where any input value less than zero is set to zero, that is,

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \qquad (1)$$

Max-Pooling Layer: This layer is tagged *maxPooling2dLayer*. Max- and average-pooling layers follow the convolutional layers for down-sampling, hence, reducing the number of connections to the following layers (usually a fully connected layer). They do not perform any learning themselves, but reduce the number of parameters to be learned in the following layers. They also help reduce overfitting. A max-pooling layer returns the maximum values of rectangular regions of its input. The size of the rectangular regions is determined by the poolSize argument of maxPoolingLayer.

Fully Connected Layer: This layer is tagged *fullyConnectedLayer*. As the name suggests, all neurons in a fully connected layer connect to all the neurons in the previous layer. This layer combines all of the features (local information) learned by the previous layers across the image to identify the larger patterns [11]. For classification problems, the last fully connected layer combines the features to classify the images. This is the reason that the outputSize argument of the last fully connected layer of the network is equal to the number of classes of the data set.

Soft Max-Layer: This layer is tagged *softmaxLayer*. Since this study seek to solve s classification problems, a softmax layer and then a classification layer must follow the final fully connected layer. The output unit activation function is the softmax function:

$$y_r(x) = \frac{\exp(a_r(x))}{\sum_{j=1}^{k}(a_j(x))} \qquad (2)$$

Where $0 \leq y_r \leq 1$ *and* $\sum_{j=1}^{k} y_j = 1$

The softmax function is the output unit activation function after the last fully connected layer for multi-class classification problems:

$$P(C_r \mid x, \theta) = \frac{P(x, \theta \mid C_r)P(C_r)}{\sum_{j=1}^{k} P(x, \theta \mid C_j)P(C_j)} = \frac{\exp(a_r(x, \theta))}{\sum_{j=1}^{k} \exp(a_r(x, \theta))} \qquad (3)$$

Where $0 \leq P(C_r \mid x, \theta) \leq 1$ and $\sum_{j=1}^{k} P(C_j \mid x, \theta) = 1$. Moreover, $a_r = \ln(P(x, \theta \mid c_r)(P(c_r))$, $P(x, \theta \mid c_r)$ is the conditional probability of the sample given class *r*, and $P(c_r)$ is the class prior probability.

Classification Layer: This layer is tagged *classificationLayer*: A classification output layer must follow the softmax layer [12]. In the classification output layer, trainNetwork takes the values from the softmax function and assigns each input to one of the *k* mutually exclusive classes using the cross-entropy function for a 1-of-*k*coding scheme

$$E(\theta) = -\sum_{i=1}^{n}\sum_{j=1}^{k} t_{ij} \ln y_j(x_i, \theta) \qquad (4)$$

Where t_{ij} is the indicator that the *i*th sample belongs to the *j*th class, θ is the parameter vector. $y_j(x_i, \theta)$ is the output for sample *i*, which in this case, is the value from the softmax function. That is, it is the probability that the network associates the *i*th input with class *j*, $P(t_j=1 \mid x_i)$.

III. RESULTS

The model was implemented on Matlab R2018a environment using the developed framework. In order to train the developed model, 10,500 samples were used as the training datasets where 2100 samples were used as the testing dataset. The training of the developed model was conducted using 30 Epoch, at 164 iteration per epoch while the total iteration is 4920 iterations. Also, the training period was calculated using the Matlab tic toc function and it was estimated

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 5, Issue 11, ISSN No. 2455-2143, Pages 57-61
Published Online March 2021 in IJEAST (http://www.ijeast.com)

to 11296 minutes 41 seconds (see figure 3). Thus, the model yields the network accuracy of 100% at the final iteration while the accuracy of the test set is 97.97%, F1 score of 0.9800, with 0.9803 Precision and Recall value of 0.9797.

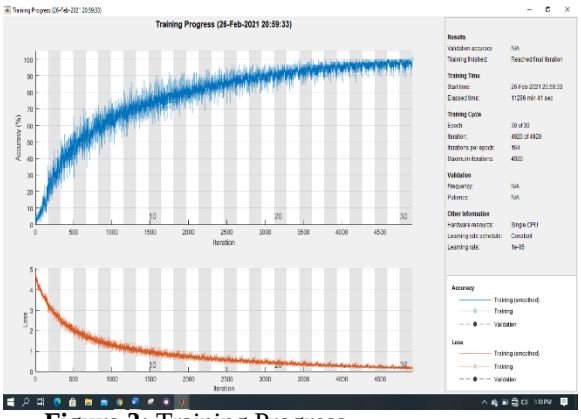

Figure 3: Training Progress

IV. CONCLUSION

This study developed a deep convolutional network called YORÙBÁNET for detection of Yorùbá alphabets, putting all factors and metrics into consideration, there is no doubt to the fact that this developed network is more than enough to take care of all related recognition system problems with respect to Yorùbá alphabets. Comparing to the previous approaches by the preceding scholars, this current technique is faster in recognition time and overall percentage accuracy is so much superb. The technique is simple, thus, making it convenient to integrate into any system or platform. The developed network is also suitable for transfer learning, hence, alphabets from other languages can be put to test in that regard.

V. REFERENCES

[1] Omniglot, "Yorùbá (Èdè Yorùbá)" (2021) https://omniglot.com/writing/Yorùbá.htm [Access on 15th February, 2021]

[2] Ajao, J. F. Olawuyi, D. O. and Odejobi, O. O. (2018) "Yorùbá Handwritten Character Recognition using Freeman Chain Code and K-Nearest Neighbor Classifier," Jurnal Teknologi dan Sistem Komputer, 6(2) doi: 10.14710/jtsiskom.6.2.2018 (pp.129-134).

[3] Oladele, M. O., Adepoju, T. M., Omidiora, E. O., Sobowale, A. A., Olatoke, O. A. and Ayeleso, E. C. (2017). An Offline Yorùbá Handwritten Character Recognition Using Support Vector Machine. International Conference of Science, Engineering and Environmental Technology, 2(13) (pp.95 – 103).

[4] Ibraheem, A. and Odejobi, O. (2011). A System for the Recognition of Handwritten Yorùbá Characters. AGIS Ethiopia, Obafemi Awolowo University, Ile-Ife, Nigeria. [Online]. Available: http://www.slideshare.net/aflat/a-system-for-therecognition-of-handwritten Yorùbá-characters

[5] Femwa, O. D. (2012). Development of a Writer- Independent Online, Handwritten Character Recognition System Using Modified Hybrid Neural Network Model. PhD. Thesis, Ladoke Akintola University of Technology, Ogbomoso.

[6] Oladayo, O. O. (2015). Yorùbá Language and Numerals' Offline Interpreter Using Morphological and Template Matching. TELKOMNIKA Indonesian Journal of Electrical Engineering 13(1) DOI: 10.11591/telkomnika.v13i1.6782 (pp.166 – 173)

[7] Akintola, A., Ibiyemi, T. and Bajeh, A. (2019). Evaluation of an Optical Character Recognition Model for Yorùbá Text. Annals. Computer Science Series. 17th Tome 1st Fasc. – 2019 (pp.33 – 42)

[8] Oladele, M. O., Adepoju, T. M., Olatoke, O. A. and Ojo, O. A. (2020). Offline Yorùbá Handwritten Word Recognition Using Geometric Feature Extraction and Support Vector Machine Classifier. Malaysian Journal of Computing, 5(2) (pp.504 – 514)

[9] Yekeen, S. A. and Ibiyemi, T. S. (2018). Edge Detection Algorithm for Yoruba Character Recognition. Advances in Vision Computing: An International Journal (AVC) 5(1/2/3/4) DOI: 10.5121/avc.2018.5401 (pp.1 – 9)

[10] Oni, O. J. and Asahiah, F. O. (2020) Computational Modelling of an Optical Character Recognition System for Yorùbá Printed Text Images. Scientific African 9

(2020) e00415 Elsevier B.V. on behalf of African Institute of Mathematical Sciences https://doi.org/10.1016/j.sciaf.2020.e00415

[11] MathWorks (2018) Create Simple Deep Learning Network for Classification. The MathWorks, Inc.

[12] Bishop, C. M. Pattern Recognition and Machine Learning. Springer, New York, NY, 2006.