

TRANSFER LEARNING BASED OFFLINE YORÙBÁ HANDWRITTEN CHARACTER RECOGNITION SYSTEM

OLUWASHINA OYENIRAN*¹, EBENEZER OYEBODE¹

¹*Ajayi Crowther University, Department of Computer Science, PMB 1066, Oyo Town, Oyo State, Nigeria*

Abstract: This study presents Transfer Learning-based framework through the use of AlexNet for the development of an offline Yorùbá Handwritten Character Recognition System. The system encompasses the upper and case characters of the Yorùbá language, and tonal letters that have a significant impact on the Yorùbá language. The model reported network accuracy of 82.8%, validation accuracy of 77.7%, with F1 score of 0.7795, precision of 0.7819 and Recall of 0.7771. While the average recognition time is estimated to 0.371372 seconds. Thus, the technique of deep learning has shown significant improvement when compared to other existing approaches in recognizing standard Yorùbá characters.

Keywords: deep learning, Yorùbá, handwritten, character, recognition

1. INTRODUCTION

Optical Character Recognition (OCR) technology could be deemed as a business solution for programming data extraction from a printed or handwritten document from a scanned document or image file and then converting the text into a machine-readable structure to be used for data processing like editing or searching. OCR embraces the electronic or mechanical transformation of images of printed, handwritten, and typed text into machine-encoded text, whether from a scanned manuscript, a photo of a document, a scene-photo, or subtitle text superimposed on an image [1].

Handwriting Recognition (HWR) is also known as Handwritten Text Recognition (HTR) which is a subset of OCR, a computer can accept and deduce handwritten from sources such as pictures, touch-screens, paper documents, and other devices [2]. HWR has been the concern of pattern recognition and machine learning scientists for numerous eons [3] it is a challenging task because individual handwritten style is exclusive, thus making it difficult for the computer to transform it into digital format [4]. It is very significant to impart computers with human handwritings which will enable computers to be able to comprehend human interactions uniquely. Moreover, it will enable computers to read postal addresses, bank cheque amounts, and forms [5]. Significantly, HWR plays a fundamental part in digital libraries by allowing the entry of image textual information into computers by digitization, image restoration, and recognition methods [6], other applications involve reading aid for the visually impaired, library automation, language processing, and multimedia design [7].

There are two major peculiar areas in HWR, according to [6] and they are the Online Character Recognition and Offline Character Recognition platforms. Study conducted by [8] revealed that the Online Handwriting Recognition encompasses the automatic transformation of the captured script as it is written on a special digitizer or Personal Digital Assistant (PDA), here, the sensor picks up the pen-tip actions as well as pen-up/pen-down switching. The captured handwritten script is transformed into letter codes which are used for text-processing roles

* Corresponding author, email: oyeniranoluwashina@gmail.com

[6]. An Offline Handwriting Recognition encompasses the processing of a stationary representation of an article where an image of the manuscript is reaped through scanner or camera is processed [9]. Due to the brands of styles in handwriting and the un-benchmarked nature of handwritings, the drawback of Offline Handwriting Recognition is the main persistent complexity in OCR and it usually involves language-specific techniques [10].

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 [3]. Nevertheless, it was noted that there are just a few character recognition systems for Yorùbá Language [11]. In this regard, this study seeks to develop a transfer learning-based framework for offline Yorùbá Handwritten Character Recognition System.

In an attempt to develop a standard Optical Character Recognition System for Yorùbá Language, [6] uses Freeman chain code and K-Nearest Neighbor in the development of an offline Yorùbá character recognition system with a recognition accuracy of 87.7%. Authors in [7] introduced a system to recognize handwritten characters of Yorùbá upper case letters. The work presented an approach of Bayesian and Decision Tree with a recognition rate of 94.44%. Authors in [8] also presented hybrid feature extraction techniques using Geometrical and Statistical Features Handwritten Character Recognition. The model was developed to train the Neural Network using Modified Counter Propagation and Modified Back-Propagation Learning Algorithms. Authors in [8] uses Geometric and Statistical Features to extract the global and local features of characters and topological features with tolerance to variation style and distortion. The results obtained showed the learning rate parameter variation had a positive effect on the network performance and a 96% recognition rate was achieved.

Correlation and Template Matching Techniques was used to develop an OCR for the recognition of Yorùbá based texts by [12] and convert English numerals in the document to Yorùbá numerals with very high accuracy when subjected to test on various sizes of Yorùbá characters, and numerals. Support Vector Machine based Yorùbá Character Recognition System to recognize Yorùbá characters was introduced by [11]. The developed recognition system experimented with 600 handwritten images for Yorùbá characters, where 480 samples were used for training and 120 samples were used for testing. The results thus showed a training time of 45.842 seconds, a recognition rate of 76.7%, and a rejection rate of 23.3%.

Recently, the study conducted by [13] obtained handwritten characters and words from different writers using the paint application and M708 graphics tablets. The characters were used for training and the words were used for testing. Pre-processing was done on the images and the geometric features of the images were extracted using zoning and gradient-based feature extraction. The characters are divided into 9 zones and gradient feature extraction was used to extract the horizontal and vertical components and geometric features in each zone. The words were fed into a multiclass SVM classifier, least square support vector machine (LSSVM) was used for word recognition. The one vs one strategy and RBF kernel were used, and the recognition accuracy obtained from the tested words ranges between 66.7%, 83.3%, 85.7%, 87.5%, and 100%. The low recognition rate for some of the words could be as a result of the similarity in the extracted features.

There is no doubt that all the aforementioned authors and other scholars have done justice in contributing to the development of the standard Optical Character Recognition System for Yorùbá Language. Nevertheless, as good as all the developed systems were, they also have their weaknesses. Weaknesses such as low recognition accuracy and low amount of dataset for training and evaluation in the case of [11]. Also, the majority of the aforementioned authors only consider the upper-case characters, ignoring the lower case, all of the authors mentioned above also ignored the alphabet “GB” and rather see it as “G” and “B”. Some of the authors also ignored the tonal characters (such as À, Á, È, É, Í, Ì, Ò, Ó, Ù, Ú). Thus, this study seek to improve on the existing knowledge, through the inclusion of omitted character “GB”, the tonal characters and lower case characters. Hence, this study exhibits a transfer learning-based framework for the development of an offline Yorùbá Handwritten Character Recognition System. Other sections of this paper are arranged as follows. Section 2 presents the methodology of the work, results and discussion of findings are presented in sections 3, while the last section is the conclusion of the paper.

2. METHODOLOGY

In order to achieve the aim and objectives of this study, this study presents the framework as specified in Figure 1 below. In order to use the dataset to train the deep learning model, there is a need to pre-process the dataset so as conform to the network requirement. Thus, the images were resized into 227 by 227 by 3 which is the standard

input size for AlexNet Model (Figure 2). Also, the noise was removed from the image through the use of the Median Filter Algorithm, so as to filter the noisy images. The image was then sorted and arranged according to each of their labels into a repository called Yorùbá Handwritten Character Dataset (YHCD). The dataset is available on [14].

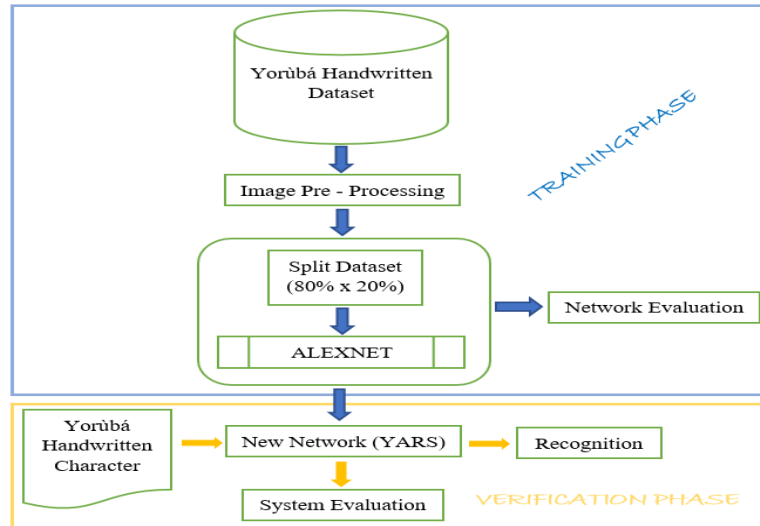


Fig. 1. System architecture.

```
>> newnet.Layers
ans =
25x1 Layer array with layers:
 1 'data'      Image Input      227x227x3 images with 'zerocenter' normalization
 2 'conv1'    Convolution      96 11x11x3 convolutions with stride [4 4] and padding [0 0 0 0]
 3 'relu1'    ReLU
 4 'norm1'    Cross Channel Normalization  cross channel normalization with 3 channels per element
 5 'pool1'    Max Pooling      3x3 max pooling with stride [2 2] and padding [0 0 0 0]
 6 'conv2'    Convolution      256 5x5x48 convolutions with stride [1 1] and padding [2 2 2 2]
 7 'relu2'    ReLU
 8 'norm2'    Cross Channel Normalization  cross channel normalization with 3 channels per element
 9 'pool2'    Max Pooling      3x3 max pooling with stride [2 2] and padding [0 0 0 0]
10 'conv3'    Convolution      384 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
11 'relu3'    ReLU
12 'conv4'    Convolution      384 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
13 'relu4'    ReLU
14 'conv5'    Convolution      256 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
15 'relu5'    ReLU
16 'pool5'    Max Pooling      3x3 max pooling with stride [2 2] and padding [0 0 0 0]
17 'fc6'      Fully Connected  4096 fully connected layer
18 'relu6'    ReLU
19 'drop6'    Dropout          50% dropout
20 'fc7'      Fully Connected  4096 fully connected layer
21 'relu7'    ReLU
22 'drop7'    Dropout          50% dropout
23 'fc'       Fully Connected  70 fully connected layer
24 'prob'     Softmax
25 'classoutput' Classification Output  crossentropyx with 'A' and 69 other classes
```

Fig. 2. Developed network architecture.

At this point, the datasets were split into two categories, 80% of which was used for enrolment and training thus it is regarded as the “Training Dataset” while the remaining 20% was used for verification, thus, it is regarded as the “Verification Dataset”. After the image pre-processing, the dataset (YHCD) was fed into the pre-trained AlexNet model. The Training dataset was split into 80% by 20%, where 80% was used for training and the remaining 20% will be used for validation automatically. After the training, the newly trained model (YARS) was saved for further analysis and future use.

At the training stage, a mini-batch gradient descent optimization algorithm is used for the learning. In mini-batch gradient descent algorithm (n) the number of training dataset samples are divided into small batches (b), then the model coefficients are updated using model error (equation (1)).

$$T = \frac{n}{b} \tag{1}$$

where T represent the total n is number of iterations per training epoch and b represent the batches. The weights (w) of CNN are optimized using error function defined in equation (2) below.

$$E_t[f(w)] = \frac{1}{b} \sum_i^{tb} = (t-1)b + 1 f(w, X_0) \quad (2)$$

where X_0 is the sample of training data and W represent the weight. At each iteration the weights are updated by rule mini batch gradient descent update rule with learning rate given in equation (3).

$$w^{t+1} = w^t - \mu \nabla_w E_t[f(w)] \quad (3)$$

3. RESULT AND DISCUSSION

This study adopted a transfer learning approach through the use of AlexNet trained Network. The model was implemented on Matlab R2018a environment using the developed framework (see Figure 1). The framework was developed by mimicking the Transfer Learning workflow but with little modifications, the modification which includes the fine-tuning of the AlexNet model, this involve changing the output of the fully connected layer of the AlexNet to conform with the new number of classes (70), also, the output layer which is the classification layer was also altered so as to conform with the new number of classes (70).

In order, to train the developed model, 12,600 training datasets were split into two parts, where the first part which comprises of 10500 sample was used for training, and the second part which contain 2100 sample was use for evaluation. The training of the developed model was conducted using 10 Epoch, at 131 iterations per epoch while the total iteration is 1310 iterations (Figure 3). Also, the training period was calculated using the Matlab tic toc function and it was estimated to 3278 minutes 24 seconds. Thus, the model yields a network accuracy of 82.81% at the final iteration while the validation accuracy of the test set is 77.71% (based on the 10500 samples), with F1 score of 0.7795, Precision of 0.7819 and Recall of 0.7771.

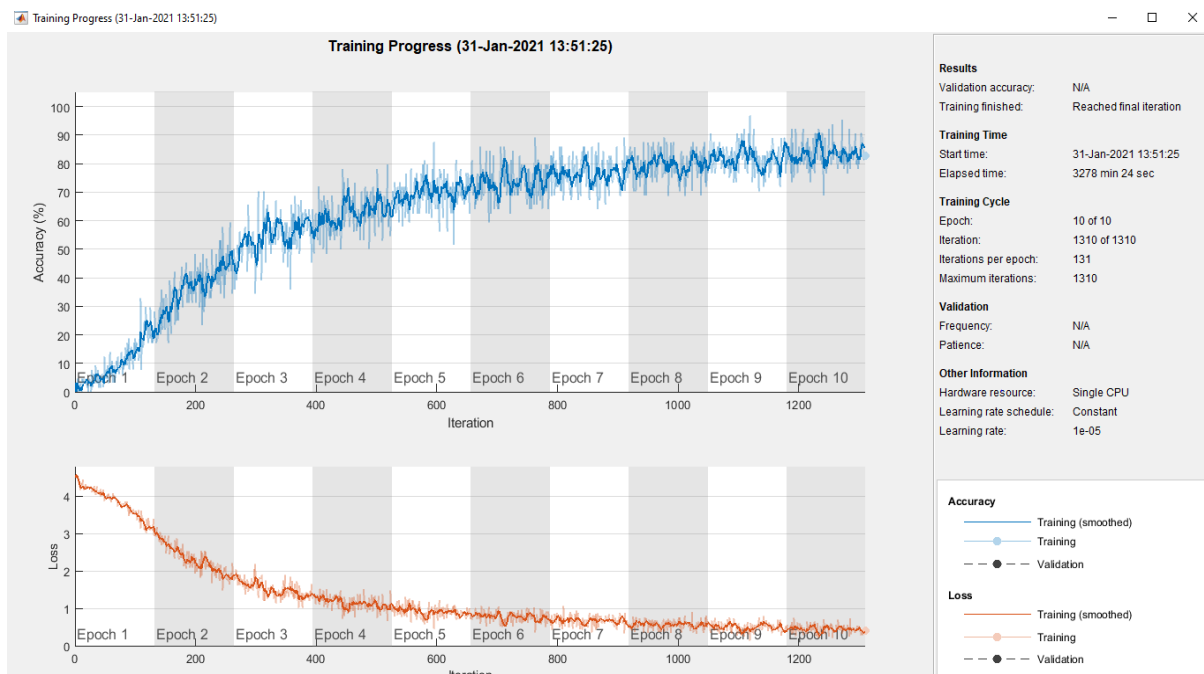


Fig. 3. Training process.

To standardize the performance of the developed system, there is a need to subject it to further tests, under standard metrics like recognition rate, rejection rate, and average recognition time. In that regard, the reserved 2100 samples of the dataset were tested manually, by subjecting each image to the developed model one after the other, the results of which is reported in Tables 1a and 1b. Hence, the recognition accuracy of the developed system was

recorded to be 91.4% and the rejection rate of 8.6% while the average recognition time is estimated to 0.371372 seconds. Thus, there is no doubt to the fact that this developed system is more appropriate and performs better.

Table 1a. Designed model evaluation using the verification dataset (upper case).

CHARACTER	TESTING SAMPLES	CORRECTLY RECOGNISED	INCORRECTLY RECOGNISED	NOT RECOGNISED	RECOGNITION RATE
A	30	29	01	0	96.7%
À	30	24	06	0	80%
Á	30	30	0	0	100%
B	30	21	09	0	73.3%
D	30	24	06	0	80%
E	30	27	03	0	90%
È	30	24	06	0	80%
É	30	18	12	0	60%
Ē	30	24	06	0	80%
F	30	29	01	0	96.7%
G	30	24	06	0	80%
GB	30	30	0	0	100%
I	30	24	06	0	80%
Ì	30	30	0	0	100%
Í	30	30	0	0	100%
H	30	30	0	0	100%
J	30	21	09	0	70%
K	30	30	0	0	100%
L	30	29	0	01	96.7%
M	30	30	0	0	100%
N	30	29	01	0	96.7%
O	30	30	0	0	100%
Ò	30	29	01	0	96.7%
Ó	30	28	02	0	93.3%
Ô	30	26	04	0	86.7%
P	30	30	0	0	100%
R	30	30	0	0	100%
S	30	30	0	0	100%
Ş	30	0	0	30	0%
T	30	30	0	0	100%
U	30	29	01	0	96.7%
Û	30	30	0	0	100%
Ú	30	26	04	0	86.7%
W	30	29	01	0	96.7%
Y	30	30	0	0	100%
SUB TOTAL	1050	935	84	31	

Nevertheless, the findings of this study with respect to Tables 1a and 1b indicate that the alphabet Ş and ş are the only characters with 0% recognition accuracy, during the verification stage, those characters always return empty space at the result panel but return the symbol at the Matlab Command Window. Since the symbol is not a Yorùbá alphabet, it is therefore recorded as not recognized, hence lead to the 0% accuracy for those characters. Further testing, through copying and pasting the symbol into the text editor automatically changes the symbol back to Ş and ş, but since it does not appear as expected at the developed application domain it cannot be recorded to be recognized.

Table 1b. Designed model evaluation using the verification dataset (Lower Case).

CHARACTER	TESTING SAMPLES	CORRECTLY RECOGNISED	INCORRECTLY RECOGNISED	NOT RECOGNISED	RECOGNITION RATE
a	30	28	02	0	93.3%
à	30	29	01	0	96.7%

á	30	25	05	0	83.3%
b	30	30	0	0	100%
d	30	29	01	0	96.7%
e	30	28	02	0	93.3%
è	30	30	0	0	100%
é	30	28	02	0	93.3%
ẹ	30	30	0	0	100%
f	30	28	02	0	93.3%
g	30	29	01	0	96.7%
gb	30	30	0	0	100%
i	30	28	02	0	93.3%
ì	30	28	02	0	93.3%
í	30	30	0	0	100%
h	30	28	02	0	93.3%
j	30	30	0	0	100%
k	30	29	01	0	96.7%
l	30	30	0	0	100%
m	30	30	0	0	100%
n	30	30	0	0	100%
o	30	30	0	0	100%
ò	30	29	01	0	96.7%
ó	30	27	03	0	90%
ọ	30	29	01	0	96.7%
p	30	30	0	0	100%
r	30	30	0	0	100%
s	30	30	0	0	100%
ş	30	0	0	30	0%
t	30	29	01	0	96.7%
u	30	30	0	0	100%
ù	30	27	03	0	90%
ú	30	26	04	0	86.7%
w	30	30	0	0	100%
y	30	30	0	0	100%
TOTAL	2100	1919	120	61	91.4%

4. CONCLUSIONS

This study presented a novel approach for recognizing Yorùbá Handwritten Character using AlexNet Pre-trained model. The framework was developed by mimicking the Transfer Learning workflow. This study ensure the inclusion of omitted character “GB”, the tonal characters (À, Á, È, É, Í, Ì, Ò, Ó, Ù, Ú, à, á, è, é, í, ì, ò, ó, ù, ú) and lower case characters, which are missing in the previous studies by various scholars. Hence, reporting network accuracy of 82.8% and the validation accuracy of 77.7%, with F1 score of 0.7795, Precision of 0.7819 and Recall of 0.7771. Also, the recognition rate of the developed system was recorded to be 91.4% and the rejection rate of 8.6% while the average recognition time is estimated to 0.371372 seconds. Based on the aforementioned facts, it is therefore concluded that Transfer Learning is appropriate for the development of Offline Yorùbá handwritten Character Recognition System.

REFERENCES

- [1] Akintola, A., Ibiyemi, T., Bajeh, A., Evaluation of an optical character recognition model for Yorùbá text, Annals Computer Science Series, 17th Tome 1st Fascicula, 2019, p. 33 – 42.
- [2] Kader, M., Deb, K., Neural network-based English alphanumeric character recognition, International Journal of Computer Science, Engineering and Applications, vol. 2, no. 4, 2012, p. 1 - 10.
- [3] Ojumaha, S., Sanjay, M., Adewole, A., Database for handwritten yorùbá characters, Ed. Springer Singapore, 2018.

- [4] Oladele, M.O., Adepoju, T.M., Olatoke, O.A., Ojo, O.A., Offline yorùbá handwritten word recognition using geometric feature extraction and support vector machine classifier, *Malaysian Journal of Computing*, vol. 5, no. 2, 2020, p. 504 – 514.
- [5] Jayadevan, R., Kolhe, S.R., Patil, P.M., Pal, U., Offline recognition of devanagari script: A survey, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* vol. 41, no. 6, 2011, p.782 – 796.
- [6] Ajao, J.F., Olawuyi, D.O., Odejebi, O.O., Yorùbá handwritten character recognition using freeman chain code and k-nearest neighbor classifier, *Jurnal Teknologi dan Sistem Komputer*, vol. 6, no. 2, 2018, p. 129-134.
- [7] Ibraheem, A., Odejebi, O.A., A system for the recognition of handwritten Yorùbá characters, AGIS Ethiopia, <http://www.slideshare.net/aflat/a-system-for-therecognition-of-handwritten-Yorùbá-characters> (22.01.2021).
- [8] Femwa, O.D., Development of a writer- independent online, handwritten character recognition system using modified hybrid neural network model, PhD. Thesis, Ladoke Akintola University of Technology, Ogbomosho, 2012.
- [9] Ajao, J.F., Olabiyisi, S.O., Omidiora, E.O., Okediran, O.O., Hidden markov model approach for offline yorùbá handwritten word recognition, *British Journal of Mathematics and Computer Science*, vol. 18, no. 6, 2016, p. 1-2.
- [10] Manoj, S., Narendra, S.A., A survey on handwritten character recognition techniques for english alphabets, *Advances in Vision Computing: An International Journal*, vol. 3, no. 1, 2012, p. 1-12.
- [11] Oladele, M.O., Adepoju, T.M., Omidiora, E.O., Sobowale, A.A., Olatoke, O.A., Ayeleso, E.C., An offline yorùbá handwritten character recognition using support vector machine, *International Conference of Science, Engineering and Environmental Technology*, vol. 2, no. 13, 2017, p. 95 – 103.
- [12] Olakanmi, O.O., Yorùbá language and numerals' offline interpreter using morphological and template matching, *Telkomnika Indonesian Journal of Electrical Engineering*, vol. 13, no. 1, 2015, p.166 – 173.
- [13] Oladele, M.O., Adepoju, T.M., Olatoke, O.A., Ojo, O.A., Offline yorùbá handwritten word recognition using geometric feature extraction and support vector machine classifier, *Malaysian Journal of Computing*, vol. 5, no. 2, 2020, p.504-514.
- [14] <https://github.com/Oluwashina90/YORÙBÁ-HANDWRITTEN-CHARACTER-DATABASE> (22.01.2021).