Handwritten Amharic Character Recognition Using a Convolutional Neural Network

Mesay Samuel Gondere, Lars Schmidt-Thieme, Abiot Sinamo Boltena and Hadi Samer Jomaa

Abstract Amharic is the official language of the Federal Democratic Republic of Ethiopia. There are lots of historic Amharic and Ethiopic handwritten documents addressing various relevant issues including governance, science, religious, social rules, cultures and art works which are very rich indigenous knowledge. The Amharic language has its own alphabet derived from Ge'ez which is currently the liturgical language in Ethiopia. Handwritten character recognition for non Latin scripts like Amharic is not addressed especially using the advantages of state-of-the-art techniques. This research work designs for the first time a model for Amharic handwritten character recognition using a convolutional neural network. The dataset was organized from collected sample handwritten documents and data augmentation was applied for machine learning.

Mesay Samuel Gondere
Arba Minch University, Faculty of Computing and Software Engineering
✉ mesay.samuel@amu.edu.et

Lars Schmidt-Thieme · Hadi Samer Jomaa
Information Systems and Machine Learning Lab, 31141 Hildesheim, Germany
✉ schmidt-thieme@ismll.uni-hildesheim.de
✉ hsjomaa@ismll.uni-hildesheim.de

Abiot Sinamo Boltena
Ethiopian Institute of Technology-Mekelle, School of Computing
✉ abiotsinamo35@gmail.com

ARCHIVES OF DATA SCIENCE, SERIES A
(ONLINE FIRST)
KIT SCIENTIFIC PUBLISHING
Vol. 6, No. 1, 2020

DOI: 10.5445/KSP/1000098011/09
ISSN 2363-9881

The model was further enhanced using multi-task learning from the relationships of the characters. Promising results are observed from the later model which can further be applied to word prediction.

1 Introduction

The Amharic language is the official language of the federal government of Ethiopia and other regional states in Ethiopia like Southern Nations, Nationalities, and People Region (SNNPR). It is a Semitic language with its own scripts where other languages of the same family in Ethiopia share the fonts. The Amharic language is believed to be derived from Ge'ez, the liturgical language of Ethiopia. The total number of characters of Ethiopic alphabets including Amharic is 446, 20 numerical representations, 9 punctuations, 8 tonal marks, 3 combining marks and 6 special characters, summed to a total of 492 characters. The Amharic alphabet, as shown in Figure 1, has 265 characters including 27 labialized and 34 base characters which are also followed by six derived characters representing derived vocal sounds of the base character (Assabie and Bigun, 2011; Weldegebriel et al., 2018). Each character represents a consonant + vowel sequence, but the basic shape of each character is determined by the consonant, which is modified by the vowel. There are lots of historic Amharic and Ethiopic handwritten documents addressing various relevant issues including governance, science, religious, social rules, cultures, and art works which are rich indigenous knowledge. However, these handwritten documents are not electronically available to be accessed and processed by the wider public using internet and emerging computing technologies (Weldegebriel et al., 2018; Meshesha and Jawahar, 2007). Optical Character Recognition (OCR) is a technology that enables the conversion of different types of written documents, such as scanned paper documents, PDF files or images into editable and searchable data. Basically OCR targets typewritten text, one glyph or character at a time. However, intelligent character recognition (ICR) targets handwritten print-script or cursive text one glyph or character at a time, usually involving machine learning.

	ä/e [ə]	u	i	a	ē	ə [ɨ], ∅	o	ʷä/ue [ʷə]	ʷa/ua
ɦ	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ		
l	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ		ሏ
ħ	ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ		
m	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ		ሟ
ɬ	ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ		
r	ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ		ሯ
s	ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ		ሷ
ʃ	ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ		ሿ
q'	ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ	ቈ	ቋ
b	በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ		ቧ
β	ቨ	ቩ	ቪ	ቫ	ቬ	ቭ	ቮ		
t	ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ		ቷ
tʃ	ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ		ቿ
χ	ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ	ኈ	ኋ
n	ነ	ኑ	ኒ	ና	ኔ	ን	ኖ		ኗ
ɲ	ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ		ኟ
ʔ	አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ		ኧ
k	ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ	ኰ	ኳ
x	ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ		
w	ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ		
ʕ	ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ		
z	ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ		ዟ
ʒ	ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ		ዧ
j	የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ		
d	ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ		ዷ
dʒ	ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ		ጇ
g	ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ	ጐ	ጓ
t'	ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ		ጧ
tʃ'	ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ		ጯ
p'	ጰ	ጱ	ጲ	ጳ	ጴ	ጵ	ጶ		
ts'	ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ		ጿ
ƛ'	ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ		
ɸ	ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ		ፏ
p	ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ		

Figure 1: The Amharic Alphabet: 34 row classes, 9 column classes. E.g. the character indicated by the arrow is the 13th label (counting left to right) with 2nd row and 6th column.

With regard to Amharic OCR and ICR research, still less has been done and the recognition techniques employed traditional approaches. Research results also show that the performance of the available prototypes are low, especially for various quality types of image and different types of fonts. The large number of alphabets, similarity of the characters, unavailability of a corpus, and the lack of a standard for Amharic fonts are mentioned to be the major reasons that complicated the efforts so far (Assabie and Bigun, 2011; Meshesha and Jawahar, 2007).

Basically character recognition includes the following phases: Pre-processing, segmentation, feature extraction, and classification. While each of the various stages has impact on the recognition accuracy, the feature extraction technique plays the major role (Hirwani and Gonnade, 2014; Purohit and Chauhan, 2016). In this regard, convolutional neural networks (CNN) have the potential to preserve detailed features including the dimensional information. Hence, offline handwritten character recognition systems have achieved successful results with the contribution of convolutional neural networks (Xiao et al., 2017; Pradeep et al., 2010). In handwritten character processing systems, due to some domain artifacts it is difficult to design a generic system which can process handwritten characters for all kinds of languages (Purohit and Chauhan, 2016). Hence, the design of recognition systems for other languages will open a way to look to the characteristics of other scripts. The contribution of this paper includes

1. addressing Amharic handwritten character recognition using a convolutional neural network for the first time,
2. pointing out the advantages of multi-task learning which can be implied from the relationships of the Amharic characters, and
3. exploring the recognition pattern of the various tasks related to the Amharic alphabet which will help in addressing the problem and indicating scalability possibilities to other scripts.

2 Related Works

Handwritten character recognition for non-Latin scripts is still an active area of research. While convolutional neural networks are the state-of-the-art techniques applied in most image recognition tasks, the recent research efforts in handwritten recognition seems to fall in two categories. The first group emphasizes the improvement of recognition accuracies by trying possible deeper and more complex architectures (Elleuch et al., 2016; Roy et al., 2017). In the opposite direction, there are attempts to simplify the architectures stressing more on the reduction of run time and space complexity of the proposed solutions (Xiao et al., 2017; Zhang et al., 2017). In their literature survey Purohit and Chauhan (2016) outlined the importance of feature extraction techniques in character recognition and revealed the need to address the enhancement of algorithms and recognition rates. Rosyda and Purboyo (2018) discussed the different methods used to address the challenges of handwritten character recognition and reported a CNN to be the best method in terms of getting higher accuracy. As a typical example of the advantages of CNNs, El-Sawy et al. (2017) implemented a CNN for Arabic handwritten characters with different parameter optimization methods to increase the performance of the CNN. The authors used two CNN layers with 80 and 64 feature maps, two pooling layers and one fully connected layer and reported a promising result of 94.9 % classification accuracy rate on testing images.

There is limited research on Ethiopic character recognition. The possible reasons mentioned are the use of a large number of characters in the alphabet, the existence of a large set of visually similar characters and the unavailability of a standard dataset until recently (Assabie and Bigun, 2011; Meshesha and Jawahar, 2007). The first attempt for Amharic offline character recognition was reported by Cowell and Hussain (2003). They approached the problem using template and signature template matching. Assabie and Bigun (2008) have implemented offline handwritten character recognition for Ethiopic script based on the characteristics of primitive strokes that make up characters. The authors also develop a comprehensive dataset for related research works. A work by Weldegebriel et al. (2018) addressed deep learning for Ethiopian Ge'ez script optical character recognition and demonstrated the promises of applying convolutional neural networks for Ethiopic scripts. A recent related parallel work by Belay et al. (2019) proposes a CNN based approach for Amharic character image recognition. The later two papers focus on frequent Amharic characters and used synthetic printed characters.

3 Methodology

This section outlines how dataset preparation and techniques of recognition were employed to undertake the study.

3.1 Dataset Preparation

The dataset for this study was organized from the dataset developed by Assabie and Bigun (2009), a comprehensive Dataset for Ethiopic Handwriting Recognition. A subset of this dataset contains offline isolated characters freely written by several participants, each participant writing the 265 Amharic characters in one page. Twelve unique handwritings were extracted as stratified samples for each 265 Amharic language characters from this dataset. The ratio 9:2:1 was applied for training, validation and test splits per alphabet/ character. Due to the importance of big dataset in machine learning (Wigington et al., 2017), the subsets representing each character were further augmented using data augmentation techniques including −15 to 15 degree random rotations, random noise, and 70–87 % resizing (diminishing). Accordingly, 1,192,500 images (4500 per character), 212,000 images (800 per character), and 106,000 images (400 per character) were used for the training, validation, and test sets respectively. Finally, the dataset was represented in numpy array format incorporating one-hot encoding for the labels and hence ready for input to the CNN model. Figure 2 shows sample characters of the dataset.

Figure 2: Sample handwritten characters from the dataset.

3.2 Convolutional Neural Network Architecture

Convolutional neural networks (CNN) are a class of deep neural networks widely used as the state-of-the-art technique in computer vision. CNNs have demonstrated the potential of automatically preserving salient features from the input and hence are not sensitive to variations. The CNN network basically is structured as a set of layers including convolution layers, sub-sampling layers and fully connected layers. From an $M \times M \times C_1$ input neuron nodes which will be convoluted with $N \times N \times C_1$ filter and stride of one, the convolution layer outputs $(M - N + 1) \times (M - N + 1) \times C_2$. The sub-sampling layers like max-pooling reduce the dimensionality of each feature map while retaining the relevant information. Finally, the fully connected layer is attached with other classifiers outputs the predictions. The rest of the CNN working concept is similar to any other neural networks including the activation functions, the loss function, regularizations, and hyper-parameter tunings. This study adapted the CNN architecture from the work of El-Sawy et al. (2017) which was designed for Arabic handwritten character recognition. Hence, the architecture shown in Figure 3 was reconstructed through incremental experiments to avoid the high-level over-fitting in El-Sawy et al. (2017) by reducing the number of channels in the convolution layers.

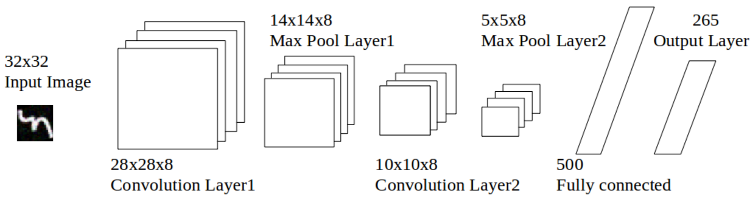


Figure 3: The proposed CNN architecture for Amharic handwritten characters.

3.3 Multi-task Learning

Multi-task learning is learning with auxiliary tasks to help improve upon the main task. Technically, it is optimizing more than one loss function in contrast to single-task learning. Multi-task learning has been used successfully across all applications of machine learning as it improves generalization by leveraging the domain-specific information contained in the training signals of related tasks. The widely used multi-task learning approach is hard parameter sharing where the hidden layers between all tasks are shared while keeping several task-specific output layers. Hence the more number of tasks, the more generalization of the main task (Ruder, 2017). From the structure of the Amharic alphabet, one can identify two related tasks. These are the row class prediction (1–34) and the column class prediction (1–9) of the alphabet. Accordingly, for this study these tasks are added by weighting each loss

$$\text{Loss} = \underbrace{\alpha_1 \cdot l\,(y, \widehat{y})}_{\text{character prediction loss}} + \overbrace{\alpha_2 \cdot l\,(y_2, \widehat{y_2})}^{\text{row prediction loss}} + \underbrace{\alpha_3 \cdot l\,(y_3, \widehat{y_3})}_{\text{column prediction loss}} \tag{1}$$

with hard parameter sharing as shown in Figure 4 and where $\alpha_1, \alpha_2, \alpha_3$ are hyper parameters introduced to control the effects of the corresponding losses.

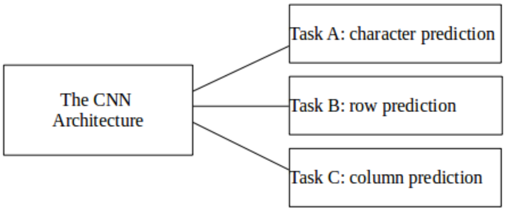


Figure 4: Multi-task learning (hard parameter sharing) using the rows and columns class.

4 Experimental Results

All the experiments were performed using TensorFlow on nVidia GPU nodes connected to the computing cluster at the Information Systems and Machine Learning Lab (ISMLL, University of Hildesheim).

4.1 Convolutional Neural Networks

The convolutional neural network has been trained with the following hyper parameter settings: A batch size of 100, a learning rate of 0.0001, L2 regularization of 0.01, and a keeping probability for dropout of 0.3. As there are several hyper parameters, the choice of the hyper parameters is empirical with a focus on the learning behavior of the model. The experiments were controlled by early stopping when the loss values show no more reduction.

As shown in Figure 5 the model converges after 200 epochs and the loss curves show a drop from 9.01 to 0.67 and from 8.81 to 2.13 for training and validation sets respectively. On the other hand, 87.48 % and 52.15 % accuracy were reached with 300 epochs for training and validation set respectively. Similarly, 2.05 loss and 52.82 % accuracy were achieved with the test set. This is an average performance attained with CNN without the need for any feature extraction technique. However, the large gap between the training accuracy of 87.48 % and the test accuracy of 52.15 % might be due to the limited number (9) of unique handwritings before data augmentation. Even though data augmentation was used to address the varieties in handwriting and helped, it may not encompass the natural varieties to scale to new datasets.

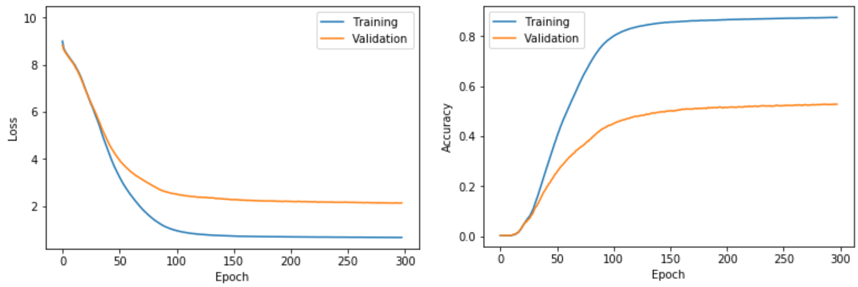
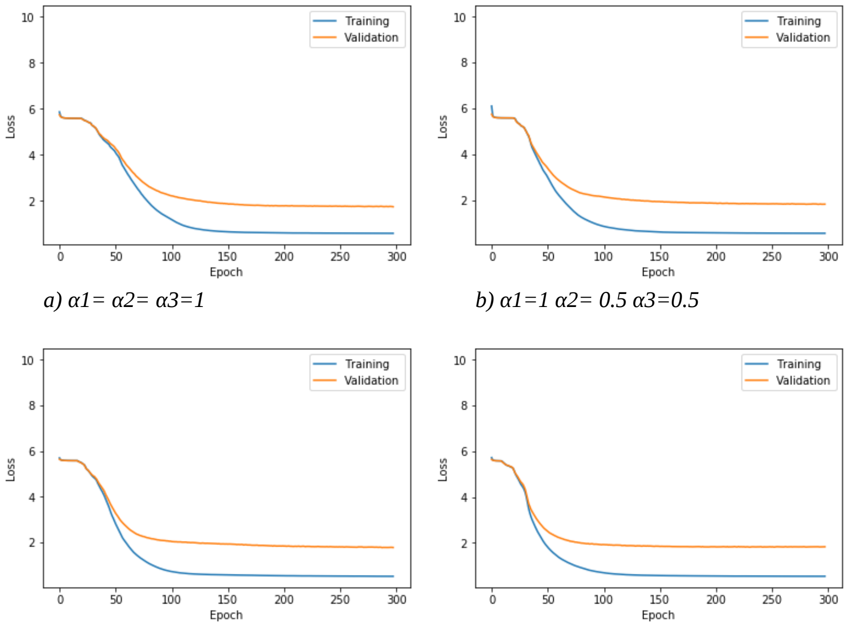


Figure 5: The loss (left) and accuracy (right) curves of the CNN model.

4.2 Multi-task Learning with Different Alpha Values

The second phase of the experiment introduced multi-task learning with the same hyper parameter settings for the CNN and different weights of alpha. Learning the auxiliary tasks (predicting the rows and columns of the alphabets) has helped in learning the main task (predicting the individual alphabet labels). Different coefficients of alpha values were used in these experiments to identify the degree of influence of each task. The results are shown in Figure 6 (a–d).

As it can easily be observed in Figure 6, a considerable improvement was attained in minimizing the loss with the multi-task learning experiments. In all the cases the loss drops from [6.10 - 5.70] to [0.58 - 0.52] and [5.74 - 5.64] to [1.83 - 1.74] for training and validation sets, respectively. This finding serves as an empirical evidence to show the support of multi-task learning in improving the generalization capability of the model. The overall comparison among all the experiments is shown in Figure 7.



a) α1= α2= α3=1

b) α1=1 α2= 0.5 α3=0.5

c) α1= 1 α2= 0.65 α3= 0.35

d) α1= 1 α2= 0.35 α3=0.65

Figure 6: The loss curves for multi-task learning with different alpha values (a–d).

From Figure 7, it was observed that the model with alpha values ($\alpha_1 = 1, \alpha_2 = 0.35, \alpha_3 = 0.65$) has performed best and showed the fastest convergence. This has implied the significance of learning the columns in supporting the main task. Accordingly a closer investigation was made to examine how auxiliary tasks (predicting the rows and columns of the alphabet) have been learned together with the main task (predicting the individual alphabet labels). This is shown in Figure 8.

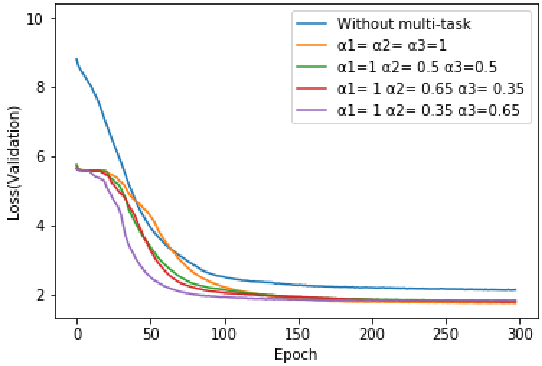
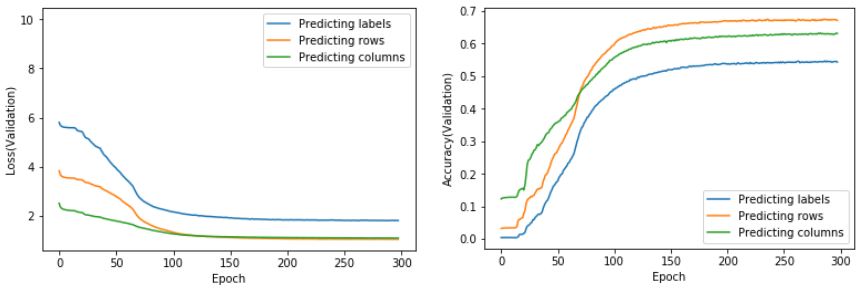


Figure 7: The overall comparison among all the experiments with (a–d) and without multi-task learning.



a) Loss curve on validation set

b) Accuracy curve on validation set

Figure 8: Examining the learning behavior of the auxiliary tasks ($\alpha_1 = 1$, $\alpha_2 = 0.35$, $\alpha_3 = 0.65$).

Figure 8 a) shows how the loss drops for all the three tasks during the multi-task learning on validation set. Accordingly, the model predicted better for the auxiliary tasks. The loss dropped from 3.83 to 1.05 and from 2.51 to 1.09 for predicting the rows and columns respectively. Similarly, 54.35 %, 67.13 %, and 63.16 % accuracies are attained in predicting labels, rows, and columns respectively. The model shows to classify columns easily in the earlier epochs. This corresponds to the nature of characters under the same column which exhibit a consistent structure over the base characters as shown in Figure 1. Finally, accuracies of 54.92 %, 68.09 %, and 65.26 % are achieved on the test set for predicting the labels, rows, and columns, respectively.

Apart from getting the improvement from the relationship of characters as placed in rows and columns, it was observed that predicting these auxiliary tasks is easier. Hence, these results will open a way to use these relationships for predicting labels and also predicting Amharic words.

5 Conclusion and Future Work

In this study Amharic handwritten character recognition was addressed using a convolutional neural network. Without the need for hand crafted feature extraction it was observed that one can achieve a good recognition result. More importantly, the relationship among the characters as placed in the Amharic alphabet has opened a way for multi-task learning. The result of the study demonstrated the relevance of these auxiliary tasks in improving the recognition accuracy. Particularly, the trained model performed better at predicting the rows and columns of the alphabets. Hence, drawing advantage of the row and column information, Amharic word predictions will be investigated in future work. A further investigation on the scalability of the proposed technique for multi-script recognition will also be explored in the future. In this study, the limitation of an unique handwritten dataset affected the performance of the models. Hence, a standard real dataset will be developed which can be used for related machine learning experiments.

Acknowledgements The authors would like to acknowledge Yaregal Assabie and Josef Bigun for providing the Ethiopic handwriting dataset.

References

Assabie Y, Bigun J (2008) Writer-independent Offline Recognition of Handwritten Ethiopic Characters. In: Proceedings of the 11th International Conference on Frontiers in Handwriting Recognition (ICFHR'08), Montreal, p. 6.

Assabie Y, Bigun J (2009) A Comprehensive Dataset for Ethiopic Handwriting Recognition. In: Bigun J, Verikas A (eds.), Proceedings SSBA'09: Symposium on Image Analysis, Halmstad University, Halmstad (Sweden), pp. 41–43. ISBN: 978-9-163339-24-0.

Assabie Y, Bigun J (2011) Offline Handwritten Amharic Word Recognition. Pattern Recognition Letters 32(8):1089–1099. DOI: 10.1016/j.patrec.2011.02.007.

Belay B, Habtegebrial T, Liwicki M, Belay G, Stricker D (2019) Factored Convolutional Neural Network for Amharic Character Image Recognition. In: IEEE International Conference on Image Processing (ICIP'19), Institute of Electrical and Electronics Engineers (IEEE), Piscataway (USA), pp. 2906–2910. DOI: 10.1109/ICIP.2019.8804407.

Cowell J, Hussain F (2003) Amharic Character Recognition Using a Fast Signature Based Algorithm. Proceedings on the 7th IEEE International Conference on Information Visualization, Piscataway (USA), pp. 384–389. DOI: 10.1109/IV.2003.1218014.

El-Sawy A, Loey M, EL-Bakry H (2017) Arabic Handwritten Characters Recognition Using Convolutional Neural Network. WSeas on Computer Research 5:9.

Elleuch M, Maalej R, Kherallah M (2016) A New Design Based-SVM of the CNN Classifier Architecture with Dropout for Offline Arabic Handwritten Recognition. Procedia Computer Science 80:1712–1723. DOI: 10.1016/j.procs.2016.05.512.

Hirwani A, Gonnade S (2014) Handwritten Character Recognition System Using Neural Network. International Journal of Advanced Research in Computer Science and Management Studies 2(2):83–88. URL: `https://www.ijser.org/researchpaper/Handwritten_Character_Recognition_using_Neural_Network.pdf`.

Meshesha M, Jawahar CV (2007) Optical Character Recognition of Amharic Documents. African Journal of Information & Communication Technology 3(2). DOI: 10.5130/ajict.v3i2.543.

Pradeep J, Srinivasan E, Himavathi S (2010) Diagonal Feature Extraction Based Handwritten Character System Using Neural Network. International Journal of Computer Applications 8(9):17–22. DOI: 10.5120/1236-1693.

Purohit A, Chauhan SS (2016) A Literature Survey on Handwritten Character Recognition. International Journal of Computer Science and Information Technologies (IJCSIT) 7(1):5. URL: `http://www.ijcsit.com/docs/Volumeijcsit2016070101.pdf`.

Rosyda SS, Purboyo TW (2018) A Review of Various Handwriting Recognition Methods. International Journal of Applied Engineering Research 13(2):1155–1164.

Roy S, Das N, Kundu M, Nasipuri M (2017) Handwritten Isolated Bangla Compound Character Recognition: A New Benchmark Using a Novel Deep Learning Approach. Pattern Recognition Letters 90:15–21. DOI: 10.1016/j.patrec.2017.03.004.

Ruder S (2017) An Overview of Multi-Task Learning in Deep Neural Networks. URL: `http://arxiv.org/abs/1706.05098`. ArXiv: 1706.05098.

Weldegebriel HT, Chen J, Zhang D (2018) Deep learning for Ethiopian Ge'ez script optical character recognision. In: 10th International Conference on Advanced Computational Intelligence (ICACI'18), Piscataway (USA), pp. 540–545. DOI: 10.1109/ICACI.2018.8377517.

Wigington C, Stewart S, Davis B, Barrett B, Price B, Cohen S (2017) Data Augmentation for Recognition of Handwritten Words and Lines Using a CNN-LSTM Network. In: 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Piscataway (USA), pp. 639–645. DOI: 10.1109/ICDAR.2017.110.

Xiao X, Jin L, Yang Y, Yang W, Sun J, Chang T (2017) Building Fast and Compact Convolutional Neural Networks for Offline Handwritten Chinese Character Recognition. Pattern Recognition 72:72–81. DOI: 10.1016/j.patcog.2017.06.032.

Zhang XY, Bengio Y, Liu CL (2017) Online and Offline Handwritten Chinese Character Recognition: A Comprehensive Study and New Benchmark. Pattern Recognition 61:348–360. DOI: 10.1016/j.patcog.2016.08.005.