**International Journal of Electrical and Computer Engineering (IJECE)**
Vol. 12, No. 4, August 2022, pp. 4164~4171
ISSN: 2088-8708, DOI: 10.11591/ijece.v12i4.pp4164-4171 

# Tifinagh handwritten character recognition using optimized convolutional neural network

**Lahcen Niharmine[1], Benaceur Outtaj[2], Ahmed Azouaoui[3]**
[1]SIME Lab, ENSIAS, Mohamed V University, Rabat, Morocco
[2]Computer Science Department, FSJE Mohamed V University, Rabat, Morocco
[3]Computer Science Department, Chouaib Doukkali University, El Jadida, Morocco

**Article Info**

***Article history:***

Received Dec 21, 2020
Revised Mar 28, 2022
Accepted Apr 18, 2022

***Keywords:***

Convolutional neural networks
Handwritten character recognition
Tifinagh alphabet

**ABSTRACT**

Tifinagh handwritten character recognition has been a challenging problem due to the similarity and variability of its alphabets. This paper proposes an optimized convolutional neural network (CNN) architecture for handwritten character recognition. The suggested model of CNN has a multi-layer feed-forward neural network that gets features and properties directly from the input data images. It is based on the newest deep learning open-source Keras Python library. The novelty of the model is to optimize the optical character recognition (OCR) system in order to obtain best performance results in terms of accuracy and execution time. The new optical character recognition system is tested on a customized dataset generated from the amazigh handwritten character database. Experimental results show a good accuracy of the system (99.27%) with an optimal execution time of the classification compared to the previous works.



***Corresponding Author:***

Lahcen Niharmine
SIME Lab, ENSIAS, Mohamed V University
Avenue Mohamed Ben Abdellah Regragui, Rabat, Morocco
Email: niharmine@gmail.com

## 1. INTRODUCTION

Tifinagh is the alphabet of the Amazigh language which is spoken widely by the North African people. The alphabet was normalized on October 17, 2001, by the Moroccan Government. This alphabet is formally admitted by the International Organization for Standardization (ISO) [1]. Tifinagh-IRCAM has 33 characters as shown in the Figure 1.

Handwritten character recognition is one of the most interesting and important area of natural language processing. Several researches have been accomplished in this domain in order to provide many important services like documents scanning, bank cheques processing, reading postal codes and different forms of handwritten documents. In the majority of previous works [2]–[4] Tifinagh handwritten character recognition system can be divided into three main phases: i) preprocessing steps: image resizing, segmentation and binarization, ii) feature extraction: this step generates features of the character image based on its geometrical characteristics, and iii) classification and recognition.

The major difficulties of different systems can be highlighted during the steps of feature extraction. Most of system have problems related to the time of training which is very high and parameters optimization of the convolutional neural network (CNN). The feature extraction requires much time and may impact the accuracy of the system. In this paper we will present a new CNN model with optimized parameters for the Tifinagh handwritten character in order to overcome the above listed difficulties. This will solve the problem of characters confusion and training time which we have faced before on previous works [2], [3].


Figure 1. Tifinagh IRCAM alphabet

The rest of this paper is formed as follows: the second section introduces a review of previous works. Then, CNN architecture is presented in the third section. The suggested new model is described in section four. In section five the experimental results are presented. Finally, we will conclude and provide suggestions for future research.

## 2. PREVIOUS WORKS

In the literature, numerous convolutional neural networks systems have been developed such as tangut character recognition based on deep learning algorithms, in this work Zhang and Han [4]. Have constructed a tangut character recognition system. They used a large training dataset which ensure a high accuracy of a deep learning system. They achieved an accuracy of 94%.

Rismiyati *et al.* [5] presented a new model of Javanese character recognition which uses deep learning techniques to classify handwritten Javanese characters. The classification used a dataset of 2,470 images from 20 characters. The size of the input image is 32×32 pixels. The classification is performed by using convolutional neural networks (CNN) and deep neural network (DNN). They obtained an accuracy of 70.22% with k-fold cross validation and 64.65% for CNN and DNN.

Zhang *et al.* [6], introduced new advances on implementing deep learning methods for handwritten Chinese character recognition and handwritten Chinese text recognition. They eliminated the need for data augmentation and model ensemble. By using deep learning methods with old approaches, they were able to achieve state-of-the-art performance for both systems.

Jindal *et al.* [7] proposed a new method based on deep convolutional neural networks. They used a dataset of 35 Gurumukhi different characters. Experimental results showed an accuracy of 98.32% for the training dataset, and 74.66% on the test data.

Tifinagh character recognition has become an active field of research in the last decade because of its introduction in the education, industry, and government institutions. Traditional systems involve many different processes including character images preprocessing, database preparation (features extraction), generation of best features and classification. Niharmine *et al.* [3] have proposed a new enhanced feature extraction based on genetic algorithms. The proposed system achieved good results with better features. The classification phase is performed using a feedforward neural network.

Ouadid *et al.* [8] presented a model built with the graph theory. They used Harris corner detector method to extract the interest points. They built the graph model representation of Tifinagh characters based on the extracted points. The classification phase was done by computing the spectral properties calculation of the adjacency matrix that represents the affinity of conformity between graphs. The system proves a recognition rate of 99.02%.

Amrouch *et al.* [9] proposed an optical character recognition (OCR) system for Tifinagh using a crossbreed approach by merging the Hough transform and hidden Markov models. After binarization, segmentation and resizing of the image, the final vector of the image character is constructed from the Hough transformation. This vector is transformed into a sequence of observations that is used for the classification and recognition phase. In the end, they apply the forward classifier to recognize the handwritten character. They obtained interesting results during the testing phase using a local database.

Oulamara and Duvernoy [10] extracted features of straight segments using the Hough transform to extract with their attributes (length and orientation). Features vectors were generated by analyzing the characters in the parametric space. The adopted method achieved interesting results during tests with the local database. Classical methods are so weak in terms of performance. Researchers have realized few projects using enhanced deep learning techniques. Tifinagh handwritten character recognition related works using deep convolutional neural networks (DCNN) are very few. Researchers have just started applying it in the last three years. Benaddy *et al.* [11] proposed a new CNN system tested on the amazigh handwritten character database (AMHCD) [12] dataset and achieved a recognition accuracy of 99.10%. The CNNs extracts features directly from raw pixels. They use a CNN system of 5 adjacent layers. The first three layers compute features extraction and the two remaining layers execute classification step.

Sadouk *et al.* [13] have developed a new system using two CNN architectures: deep belief networks (DBNs) and CNNs. The authors used the AMHCD database to train and test the two networks. Experimental tests show an accuracy of 95.47% while CNNs perform an accuracy of 98.25%.

The major issues faced by previous research are mainly: first, they need to perform preprocessing steps that requires a significant time, second, they can face problems of confusion between many characters such as : ‘Yaz’ and ‘Yazz’, ‘Yay’ and ‘Yag’, ‘Yadd’ and ‘Yatt’, third the time of recognition or classification is too much big, fourth the majority of these systems use only 31 characters instead of 33 characters. In order to resolve these difficulties and issues we will apply a new technique based on Keras neural networks library to classify and recognize Tifinagh character. The Keras is a machine learning open source code library released by François Chollet on March 27, 2015. It has been being widely used in computer vision, especially the field of pattern recognition.

## 3. THE PROPOSED METHOD

### 3.1. Convolutional neural networks

Convolutional neural networks are very similar to classical neural networks. They are composed of neurons with learning weights and prejudices. Each neuron receives some input, executes a dot product, and optionally tracks it non-linearly. The term "deep neural network" refers to the number of the term deep neural network refers to the number of hidden layers. For a normal neural network, it usually uses just one hidden layer, and deep related to multiple hidden layers. The multiple hidden layers between the raw input data and the output label allow the network to learn features at various levels of abstraction, making the network itself able to make features extraction. LeNet [14] is the first CNN built by Lecun in 1998.

Regular neural networks pickup an input vector and send it via a sequence of hidden layers. Each hidden layer is made up from a group of neurons, where each neuron is completely connected to all or any neurons within the preceding layer, and where neurons during a single layer function completely independently and don't share any connections. The last layer, which is fully connected, is named the “output layer” and in classification parameters it serves as the category scores.

Convolutional neural networks read input images directly and limit the architecture more sensitively. ConvNet layers have neurons in three dimensions: width, height, depth. The depth here refers to the third dimension of an activation volume. For a red, green, blue (RGB) type image, the depth is 3. The final output layer would have dimensions of 1×1×class since an evaluation vector for a single class is created at the end of the ConvNet architecture. The Figure 2 represents the architecture of convolutional neural networks.

### 3.2. ConvNets layers

A simple ConvNet is a sequence of layers. Every layer of a ConvNet transforms one volume of activations to a new one through a differentiable function. ConvNet architectures have three important types of layers: pooling layer, convolutional layer and fully connected layer. We will assemble these layers to construct a full CNN architecture.

For an RGB image with size 32×32 the ConvNet layers details are as follows: i) INPUT [32×32×3] will take the data pixel of the image, in CNN case the image has dimensions of 32×32 with three colors R.G.B, ii) CONV layer will calculate the output of neurons that are linked to local regions in the input, each computing a dot product between their weights and a small region they are connected to in the input volume. The result is a volume such as [32×32×k] if we want to use k filters, iii) rectified linear unit (ReLU) layer computes a threshold operation to each element of the input, like the max(0,x) thresholding at zero which makes the volume size unchanged ([32×32×k]), iv) pooling layer will compute a downsampling operation on the spatial dimensions (width, height), with output of volume like [16×16×k], and v) fully Connected layer will compute the class scores, with a volume result of size [1×1×class].

The filters k act as feature detector from the original input image. Then, a non-linearity function is then computed to the result of the convolutional operation to achieve the so-called activation map (also named feature map). Many projects and research have built different CNN architectures for classification. Some of the most important deep CNN networks are AlexNet [15], visual geometry group (VGG) networks [16], and region-based convolutional neural networks [17].

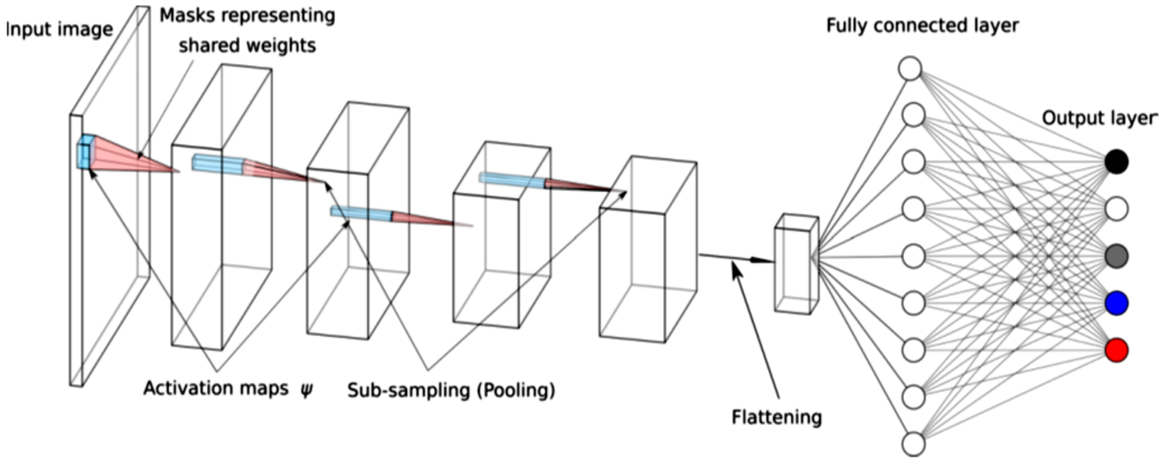


Figure 2. Convolutional neural network architecture

## 4. RESEARCH METHOD

We have used Keras library to implement our CNN model. We just add one layer at a once starting from the beginning of the CNN (the input). The first layer is a convolutional layer (Conv2D). It is composed of learnable filters. We adjust the number of filters to 32 for the two first layers and 64 filters for the two second layers and 128 filters for the last two layers. Each filter converts a part of the image using the kernel filter. The kernel filter matrix is used on the whole image. Filters can be considered as a transformation of the image. The advantage of our model is that CNN can extract features that are useful in each place from the transformed images (features map).

The second key layer in our CNN model is the pooling layer (MaxPool2D). The role of this layer is a downsampling filter. It picks the maximal value from the two neighboring pixels. This operation is computed to reduce the computational cost and lower overfitting. The pooling size is well chosen. The downsampling become important when the pooling dimension is high.

The purpose of our architecture is to associate convolutional and pooling layers, CNN are adequate and able to couple local features and extract more global features of the image. ReLU is the activation function max (0, x) as shown in (1). The activation function is used to add nonlinearity to the network. The used function is ReLU.

$$f(x) = \begin{cases} 0\ for\ x < 0 \\ 1\ for\ x \geq 0 \end{cases} \tag{1}$$

The flatten layer is used transform the final feature maps into one single vector. This flattening phase is applied in order to use a fully connected layer after convolutional/maxpool layers. It assembles all the local features of the previous convolutional layers.

Finally, we employ the features in the two fully connected (Dense) layers which is an artificial neural networks (ANN) classifier. In the last layer, a dense layer with 33 outputs and activation function SoftMax, the net outputs values of probability of each class. The CNN architecture has 291 073 trainable parameters as shown in the Table 1.

Table 1. Proposed architecture parameters

| Layer (type) | #kernels | Kernel/pool size | Output shape | #param |
|---|---|---|---|---|
| Image (InputLayer) | – | – | (1, 3, 28, 28) | 0 |
| Conv1 (Conv2D) | 32 | 5 × 5 | (32, 28, 28) | 832 |
| Conv2 (Conv2D) | 32 | 5 × 5 | (32, 28, 28) | 25 632 |
| Pool1 (MaxPooling2D) | – | 2 × 2 | (32, 14, 14) | 0 |
| Conv3 (Conv2D) | 64 | 3 × 3 | (64, 14, 14) | 18 496 |
| Conv3 (Conv2D) | 64 | 3 × 3 | (64, 14, 14) | 36 928 |
| Pool2 (MaxPooling2D) | – | 2 × 2 | (64, 7, 7) | 0 |
| Dense128 (Dense) | – | – | (128) | 204928 |
| Dense33 (Dense) | – | – | (33) | 4257 |
| Output (SoftMax) | – | – | (33) | 0 |
| Output (SoftMax) | – | – | (33) | 0 |

## 5. RESULTS AND DISCUSSION

### 5.1. Dataset preparation

We have built a new Amazigh data set similar to the modified national institute of standards and technology (MNIST) dataset format from the Amazigh handwritten character database (AMHCD). Handwritten character images were converted to 28×28 images and transformed to csv format using python Image library. The purpose of this operation is to get our data ready for training and testing by CNN using Keras library. The split of dataset is described in the Table 2.

Table 2. Dataset partition

| Partition | Number of characters |
|---|---|
| Training images (75%) | 19 968 |
| Validation images (25%) | 4 992 |
| Total | 24 960 |

The first step is to load data converted dataset images training images, testing images and labels. Labels are 33 characters from 0-32. Then we compute a grayscale normalization to reduce the effect of illumination's differences. The next step is to reshape image in 3 dimensions (height=28 px, width=28 px, canal=1). The data visualization can be performed as shown in the Figure 3:

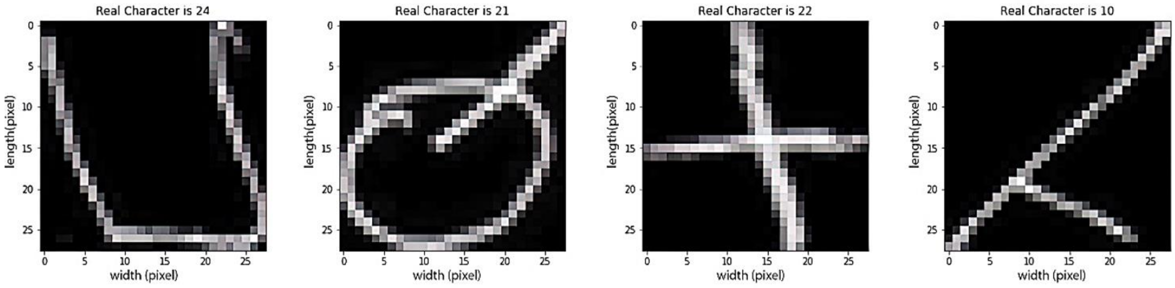


Figure 3. Visualization of character features

### 5.2. Results and discussions

The system was constructed using Keras CNN model with TensorFlow as backend. We used 60 epochs to train the DCNN. The new approach outputs the best performance without any preprocessing step (such as in [18]–[23]), The training process requires about 2,491 seconds to reach the maximum accuracy 99.37% at epoch 47 as shown in the Figure 4. We can conclude that we have achieved a very high classification accuracy and very low loss rate. Nevertheless, the prediction of some characters was wrong during the classification phase. This is due to the similarity between some character like (ⵙ and ⵔ ) and (ⴽ and ⴽⵯ).

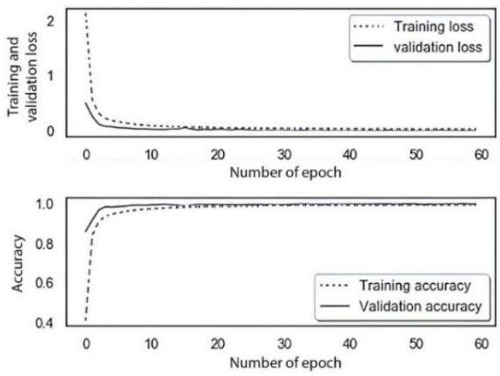


Figure 4. The accuracy and the loss of the proposed architecture (60 epochs)

We have generated the confusion matrix as shown in Figure 5. We will use it to summarize the performance of our classification algorithm. The generated matrix in Figure 5 is a summary of prediction results on our classification problem. The number of correct and incorrect predicted characters are summarized with count values and broken down by each class. As we can remark in the table summary of the matrix the number of misclassified characters is only 4, a small number compared to the predicted ones. The wrong classification is due to the format of character which is not well written during the construction of AMHCD database.

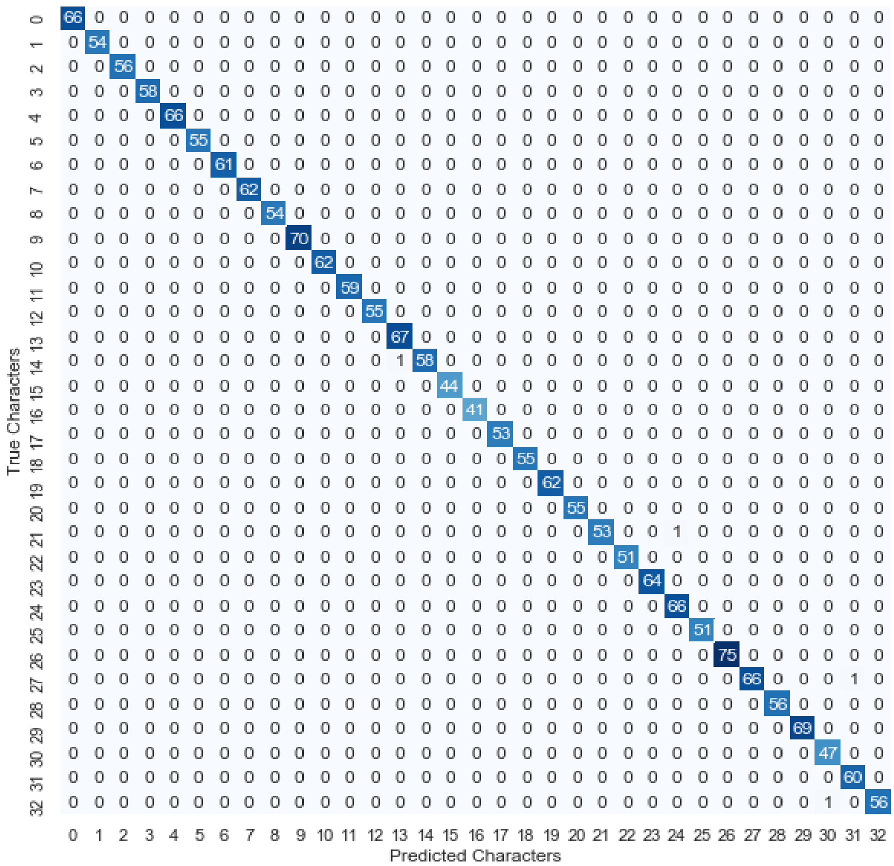


Figure 5. Confusion matrix of the proposed model

### 5.3. Optimization of the hyperparameter learning rate

In our model we have made different tests with different learning rate values. The learning rate parameter is the most crucial hyperparameter when configuring a neural network. It supervises how much to change the model in response to the estimated error each time the model weights are updated. Deciding the value of the learning rate is a difficult and challenging task. The Table 3 provides results of the proposed architecture at different learning rate. The optimal learning rate parameter is 0.009 with training accuracy 99.27%.

Table 3. Tests result based on different values of learning rate parameter

| Learning rate | Accuracy | Epoch | Time (Seconds) |
|---|---|---|---|
| 0.001 | 99.22% | 60 | 3420 |
| 0.009 | 99.27% | 38 | 1976 |
| 0.0006 | 99.29% | 50 | 2600 |
| 0.0005 | 99.37% | 47 | 2491 |
| 0.0004 | 99.37% | 57 | 3363 |
| 0.0003 | 99.35% | 58 | 3190 |
| 0.0007 | 99.23% | 48 | 2832 |
| 0.0008 | 99.31% | 44 | 2376 |

### 5.4. Comparison of the achieved results with other previous works

The proposed system has shown good result comparing it with previous works. The best training accuracy with a good training time is and 99.27% as shown in the Table 4. The use of Keras library and the optimization of the hyperparameter learning have led to build an improved character recognition system with good accuracy and very good time of execution.

Table 4. Comparison of the achieved results with other previous works

| Previous work | Number of Image used from the AMHCD | Training size | Test size | Accuracy |
|---|---|---|---|---|
| Geometrical methods [21] | 1 700 | 1000 | 700 | 92.30% |
| Baselines Features [24] | 24 180 | 21 762 | 2 418 | 94.96% |
| HMMs features [19] | 20 180 | 16 120 | 8,060 | 97,89% |
| Fusion of Classifiers Neural Networks and Support Vector Machine [20] | 165 | 33 | 33 | 81.21% |
| MLP and HMM [25] | 7200 | 1800 | 5400 | 92.33% |
| CNN DBN [13] | 24 180 | - | | 98.25% |
| CNN [11] | 25 740 | 20 592 | 5 148 | 99.10% |
| Proposed system | 25 740 | 19 305 | 6 435 | 99.27% |

## 6. CONCLUSION

In this paper, we have built a new optimal Tifinagh handwritten character recognition system based on optimized deep convolutional neural networks. The system was trained for recognizing the 33 characters using AMHCD dataset. Experimental tests are conducted with 33 class cross validation. The system outperforms all traditional works by solving issues of slowness and confusion between some characters. The experiment result shows CNN model is able to achieve the best training accuracy of 99.27%.

The Tifinagh character recognition system still has different challenging problems that need to be solved. For example, the similarity between some characters of the amazigh handwritten character database (AHCD) databases that leads to wrong prediction and the training time should be reduced. In the perspectives, we plan to improve the training time and solve confusion problems for composed characters by introducing new improvements on the AHCD database and the CNN architecture.

## ACKNOWLEDGEMENTS

I would like to express my profound gratitude to my family for their constant support which was vital in order to finish this paper and the reviewers for their valuable comments.

## REFERENCES

[1] D. Soulaimani, “Becoming amazigh: standardisation, purity, and questions of identity,” *The Journal of North African Studies*, vol. 21, no. 3, pp. 485–500, May 2016, doi: 10.1080/13629387.2016.1157481.
[2] L. Niharmine, B. Outtaj, and A. Azouaoui, “Recognition of handwritten tifinagh characters using gradient direction features,” *Journal of Theoretical and Applied Information Technology*, vol. 95, no. 13, pp. 3087–3095, 2017.
[3] L. Niharmine, B. Outtaj, and A. Azouaoui, “Tifinagh handwritten character recognition using genetic algorithms,” in *2018 International Conference on Advanced Communication Technologies and Networking (CommNet)*, Apr. 2018, pp. 1–6, doi: 10.1109/COMMNET.2018.8360267.
[4] G. Zhang and X. Han, “Deep learning based tangut character recognition,” in *2017 4th International Conference on Systems and Informatics (ICSAI)*, Nov. 2017, pp. 437–441, doi: 10.1109/ICSAI.2017.8248332.
[5] Rismiyati, Khadijah, and A. Nurhadiyatna, “Deep learning for handwritten Javanese character recognition,” in *1st International Conference on Informatics and Computational Sciences (ICICoS)*, 2017, pp. 59–64, doi: 10.1109/ICICOS.2017.8276338.
[6] X.-Y. Zhang, Y.-C. Wu, F. Yin, and C.-L. Liu, “Deep learning based handwritten Chinese character and text recognition,” in *Cognitive Computation Trends*, Springer International Publishing, 2019, pp. 57–88.
[7] U. Jindal, S. Gupta, V. Jain, and M. Paprzycki, “Offline handwritten gurumukhi character recognition system using deep learning,” in *Advances in Intelligent Systems and Computing*, vol. 1064, Springer Singapore, 2020, pp. 121–133.
[8] Y. Ouadid, B. Minaoui, and M. Fakir, “Spectral graph matching for printed tifinagh character,” in *2016 13th International Conference on Computer Graphics, Imaging and Visualization (CGiV)*, Mar. 2016, pp. 105–111, doi: 10.1109/CGiV.2016.29.
[9] M. Amrouch, Y. Es saady, A. Rachidi, M. El Yassa, and D. Mammass, “Printed amazigh character recognition by a hybrid approach based on hidden markov models and the hough transform,” in *2009 International Conference on Multimedia Computing and Systems*, Apr. 2009, pp. 356–360, doi: 10.1109/MMCS.2009.5256672.
[10] A. Oulamara and J. Duvernoy, “An application of the Hough transform to automatic recognition of berber characters,” *Signal Processing*, vol. 14, no. 1, pp. 79–90, Jan. 1988, doi: 10.1016/0165-1684(88)90045-X.
[11] M. Benaddy, O. El Meslouhi, Y. Es-saady, and M. Kardouchi, “Handwritten tifinagh characters recognition using deep convolutional neural networks,” *Sensing and Imaging*, vol. 20, no. 1, Dec. 2019, doi: 10.1007/s11220-019-0231-5.
[12] Y. E. Saady, A. Rachidi, M. El Yassa, and D. Mammass, “Amazigh handwritten character recognition based on horizontal and vertical centerline of character,” *International Journal of Advanced Science and Technology*, vol. 33, no. 17, pp. 33–50, 2011.
[13] L. Sadouk, T. Gadi, and E. H. Essoufi, “Handwritten tifinagh character recognition using deep learning architectures,” in

*Proceedings of the 1st International Conference on Internet of Things and Machine Learning*, Oct. 2017, pp. 1–11, doi: 10.1145/3109761.3109788.

[14] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998, doi: 10.1109/5.726791.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[16] S. Karen and Z. Andrew, "Very deep convolutional networks for large-scale image recognition," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2014.

[17] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142–158, Jan. 2016, doi: 10.1109/TPAMI.2015.2437384.

[18] N. Aharrane, A. Dahmouni, K. El Moutaouakil, and K. Satori, "A robust statistical set of features for amazigh handwritten characters," *Pattern Recognition and Image Analysis*, vol. 27, no. 1, pp. 41–52, Jan. 2017, doi: 10.1134/S1054661817010011.

[19] M. Amrouch, A. Rachidi, M. El Yassa, and D. Mammass, "Handwritten amazigh character recognition system based on continuous HMMs and directional features," *Int. Journal of Modern Engineering Research*, vol. 2, no. 2, pp. 436–441, 2012.

[20] M. Oujaoura, B. Minaoui, M. Fakir, R. El Ayachi, and O. Bencharef, "Recognition of Isolated Printed Tifinagh Characters," *International Journal of Computer Applications*, vol. 85, no. 1, pp. 1–13, Jan. 2014, doi: 10.5120/14802-3005.

[21] A. Djematene, B. Taconet, and A. Zahour, "A geometrical method for printed and handwritten berber character recognition," in *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, 2002, vol. 2, pp. 564–567, doi: 10.1109/ICDAR.1997.620564.

[22] O. Aiadi, A. Korichi, and M. L. Kherfi, "Recognizing Arabic handwritten words using multiple features and classifier selection," in *International Conference on Networking and Advanced Systems (ICNAS)*, 2019, pp. 1–5, doi: 10.1109/ICNAS.2019.8807871.

[23] C.-C. Chang, A. Arora, L. P. Garcia Perera, D. Etter, D. Povey, and S. Khudanpur, "Optical character recognition with Chinese and Korean character decomposition," in *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, Sep. 2019, pp. 134–139, doi: 10.1109/ICDARW.2019.40094.

[24] Y. Es Saady, M. Amrouch, A. Rachidi, M. El Yassa, and D. Mammass, "Handwritten tifinagh character recognition using baselines detection features," *International Journal of Scientific and Engineering Research*, vol. 5, no. 4, pp. 1177–1182, 2014.

[25] B. El Kessab, C. Daoui, K. Moro, B. Bouikhalene, and M. Fakir, "Recognition of handwritten tifinagh characters using a multilayer neural networks and hidden markov model," *Global Journal of Computer Science and Technology*, vol. 11, no. 5, 2011.

## BIOGRAPHIES OF AUTHORS

**Lahcen Niharmine** is a Ph.D. student within the SIME Lab at ENSIAS, University Mohamed V, Rabat, Morocco. He obtained a degree of an Engineer in Computer Science in 2009. His doctoral research investigates the use of metaheuristics in the field of Natural Language Processing. He can be contacted at email: niharmine@gmail.com.

**Benaceur Outtaj** received his license in Computer Science and Engineering and Master in Computer Science and Telecommunication from University of Mohammed V, Rabat, Morocco. His research interests are in the area of Natural Language Processing a, Artificial Intelligence, Ecommerce security and cloud computing. Dr. Outtaj was the vice president of faculty of the economy and human sciences during the period of 2016-2020. He was also a co-organizer and co-chair of many interesting conferences and events. He can be contacted at email: b.outtaj@um5r.ac.ma.

**Ahmed Azouaoui** received his license in Computer Science and Engineering in June 2001 and Master in Computer Science and Telecommunication from University of Mohammed V - Agdal, Rabat, Morocco in 2003. He received his Ph.D. in Computer Science in 2014 and Engineering at Department of Computer Science, ENSIAS (National School of Computer Science and Systems Analysis), Rabat, Morocco. Currently, he is an Associate Professor at Department of Computer Science, Faculty of sciences, University Chouaib Doukkaly, El Jadida, Morocco. His areas of interest are Information systems, Coding Theory and Artificial Intelligence. He can be contacted at email: azouaoui.a@ucd.ac.ma.