ISSN 2278-3091
Volume 9, No.1, January – February 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse26912020.pdf
https://doi.org/10.30534/ijatcse/2020/26912020

Handwritten Phoenician Character Recognition and its Use to Improve Recognition of Handwritten Alphabets with Lack of Annotated Data

Lamyaa Sadouk[1], Taoufiq Gadi[2], El Hassan Essoufi[3], Abelhak Bassir[4]
[1] Faculty of Science and Technology, University Hassan 1st, Settat, Morocco, lamyaa.sadouk@gmail.com
[2] Faculty of Science and Technology, University Hassan 1st, Settat, Morocco, gtaoufiq@yahoo.fr
[3] Faculty of Science and Technology, University Hassan 1st, Settat, Morocco, e.h.essoufi@gmail.com
[4] Faculty of Science and Technology, University Hassan 1st, Settat, Morocco, abdelhak.bassir@uhp.ac.ma

ABSTRACT

Unlike Latin, the recognition of Phoenician handwritten characters remains at the level of research and experimentation. In fact, such recognition can contribute to performing tasks such as automatic processing of Phoenician administrative records and scripts, the digitization and the safeguarding of the written Phoenician cultural heritage. As such, the availability of a reference database for Phoenician handwritten characters is crucial to carry out these tasks. To this matter, a database for Phoenician handwritten characters (PHCDB) is introduced for the first time in this paper. We also explore the challenges in the recognition of Phoenician handwritten characters by proposing a deep learning architecture trained on our database. Furthermore, we propose a transfer learning system based on Phoenician character shapes to improve the recognition performance of Tifinagh Handwritten character, and we thereby affirm the possibility of the Tifinagh alphabet being derived from the Phoenician alphabet. Finally, based on Phoenician characters, we introduce a fast, global and light-weight transfer learning system for the recognition of any alphabet which lacks annotated data.

Key words : feature extraction, pattern recognition, Phoenician character recognition, deep learning, transfer learning.

1. INTRODUCTION

Character classification is an important part in many computer vision problems such as Optical Character Recognition. Over the last several years, there have been important developments in the field of automatic recognition of characters. The main objective is to associate a symbolic representation with a sequence of graphical symbols.

Over the past few years, there has been a great interest in Phoenician scripts. According to generally accepted estimates, the corpus of Phoenician-Punic inscriptions comprises about 12,000 inscriptions from all the countries of the Mediterranean [1]. And, as noted by [2], the sheer quantity and scattered nature of the documents spread over a wide span of time have affected research and caused considerable difficulties in the knowledge, availability and use of these sources. In order to tackle this issue, one attempt to make Phoenician texts available to the academic community was the production of a collection of all Phoenician and Punic epigraphic documents in the form of a data bank, known as the CIP project [1]. But, there is still a need for the development of a system of the recognition of Phoenician handwriting characters in order to allow the automatic processing of Phoenician scripts as well as the digitization and preservation of cultural heritage of Phoenician writing. Indeed, this recognition could have several important applications, such as processing of records of land conservation or the editing of old documents. To perform this recognition, a standard reference database is required. However, up to now and to our knowledge, there exists no Phoenician handwritten alphabet dataset. To this matter, this article aims at presenting a new database for Phoenician handwritten characters (PHCDB).

Another contribution of this paper is to develop a deep learning recognition system for Handwritten Phoenician Character recognition which is based on Convolutional Neural Networks (ConvNets). These ConvNets has demonstrated state-of-the-art performance in several fields including health care [3,4] , computer vision such as face recognition [5], and character recognition such as handwritten Tifinagh recognition [6], etc.

On the other hand, knowing that the Phoenician alphabet is the ancestor of most of existing alphabets, the proposed Phoenician handwritten recognition system can further be employed for the improvement of classification/recognition of these alphabets. As such, we propose: (i) a transfer learning system which uses Phoenician characters' shapes to improve the recognition the Handwritten Tifinagh alphabet, knowing that there is a debate on the origin of this latter with some researchers maintaining it is Phoenician in origin while others claiming it is exclusively Amazigh [7]; (ii) a light-weight and fast transfer learning network for recognizing existing alphabets which experience a lack of annotated data.

In the next section, our new database of Phoenician handwritten characters is covered. Then, Section 3 describes our deep learning system for the recognition of the Phoenician handwritten alphabet. Section 4 presents our transfer learning system which uses Phoenician characters to improve the recognition of the Tifinagh alphabet. In section 5, we introduce the light-weight transfer learning system for the recognition of alphabets which lack annotated data. Finally, section 6 concludes our work.

2. HANDWRITTEN PHOENICIAN CHARACTER DATASET

2.1 Phoenician alphabet

The Phoenician alphabet, known as the oldest verified alphabet, was a Northern Semitic language used by the ancient civilization of Phoenicia in modern-day Syria, Lebanon, and northern Israel [8]. The earliest Phoenician inscription found dates from the 11th century BC (Figure 1). The Phoenician alphabet, which the Phoenicians adapted from the early West Semitic alphabet [9], is ultimately derived from Egyptian hieroglyphs [10]. It became one of the most widely used writing systems, spread by Phoenician merchants across the Mediterranean world, where it was adopted and modified by many other cultures. Indeed, it is the ancestor of the Greek alphabet and, hence, of all Western alphabets. The Paleo-Hebrew alphabet is also a local variant of Phoenician, as is the Aramaic alphabet, the ancestor of the modern Arabic.

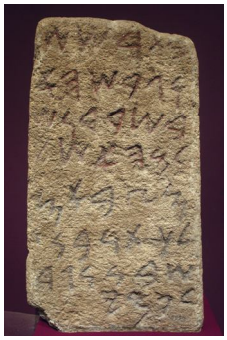


Figure 1: Phoenician text inscribed on stone, 1st millennium BCE.

The Phoenician alphabet is an alphabet of Abjad type [11], consisting of 22 consonant letters only, leaving vowel sounds implicit (Table 1).

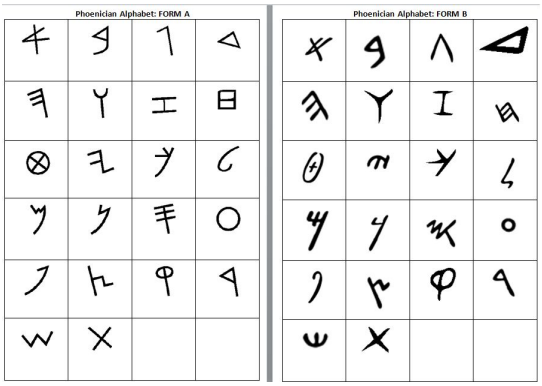
2.2 Handwritten Phoenician Character Dataset

Up to now, there exists no database of the Phoenician handwritten alphabet. In respect to the problem of the implementation of an optical character recognition (OCR) system for Phoenician handwriting scripts, we need to conduct the following tasks: (i) the collection of dataset of Phoenician handwritten characters, (ii) the pre-processing of characters.

Table 1: Phoenician alphabet with their Correspondents (names and values) in Latin Characters**.**

Name {value}	Letter(s)	Name {value}	Letter(s)
Alpeh {,}		Lameth {L}	
Beth {B}		Mem {M}	
Gimmel {G}		Nun {N}	
Daleth {D}		Samekh {S}	
He {H}		Ayin {Ap}	
Waw {W}		Pe {P}	
Zayin {Z}		Tsadi ts {So}	
Heth { χ - Ho}		Qoph {Q}	
Teth { Θ - To}		Resh {R}	
yodh{Y}		Sin {Shat}	
Kaph {K}		Taw {T}	

First, data was collected by 500 writers (authors). These authors were given one of the 2 forms (A or B) at random, where Phoenician characters of Form A have a slightly different style from those of Form B (Figure 2.a). Then they were asked to reproduce/write characters within the given form, as illustrated by Figure 2.b. Thus, 44 characters are written by each author and a total of 11,000 are gathered.



(a)

(b)

Figure 2: (a) consists of the two forms A (left) and B (right) given to writers, where each form has a different style of alphabet writing. (b) is an example of two sample forms filled by two writers.

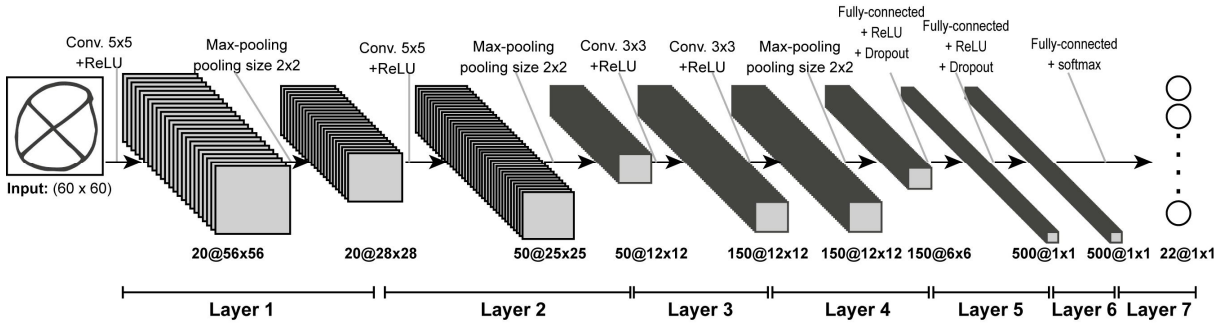


Figure 3: Architecture of the proposed ConvNet.

The forms were scanned with a quality of 300 dpi. In order to reduce the processing time and to do as much as possible automatically, a program was developed in Matlab to automatically extract each character from each filled form. After collecting and storing the images, conventional preprocessing tasks have been performed such as: noise filtering and image binarization. Also, character images have been resized to remove extra white space. Due to writing errors, a manual verification of data extracted from the collected forms was required. After removing badly written or unclear characters, we were left with 10,714 images i.e., 487 images per character. Then, images from the same class (e.g., same character) were saved into a separate folder with its own name. Figure 4 shows an example of images for the character Kaph {K} stored in our database.

Figure 4: Example of images of character Kaph {K} stored in our database.

The HPCDB database is publicly available for the purpose of research. It is available at: https://osf.io/4j9b6/

3 HANDWRITTEN PHOENICIAN CHARACTER RECOGNITION

The purpose of this section is to develop a deep learning framework for the handwritten Phoenician character recognition by training a Convolutional Neural Network (ConvNet) using the created Phoenician dataset. To do so, we first layout the ConvNet architecture and simulation settings, then describe the experiment and discuss results.

3.1 Phoenician ConvNet

A ConvNet is a concatenation of an input layer, an output layer, and multiple hidden layers. Compared with fully connected neural networks, the parameters number of these models is widely reduced by sharing weights and biases [12].

In our study, a ConvNet is fed by Phoenician images which are labeled and proportionally resized to fit a 60 × 60 frame. Its architecture is detailed in Figure 3 and a summary of its parameters is given in Table 8 (Appendix B). It is composed of seven adjacent layers. Each of first four layers consists of a stack of either Convolution, ReLU and Maxpooling units or Convolution and ReLU units, while each of the last three layers consists of a Fully Connected layer followed by a ReLU.

Training is performed using the adaptive learning rate optimization algorithm ADAM, where the learning rate and the exponential decay rates for the first and second moment estimates are set to 0.001, 0.9 and 0.99 respectively. In order to prevent overfitting of the network during training, the dropout layer is used with a rate of 0.5 and data augmentation is performed by randomly rotating, translating/shifting and jittering half of the training images. Training is conducted for 70 epochs with a batch size of 100. We report results with 3-fold cross validation and we take the average accuracy of the validation set as the final evaluation.

3.2 Experiments and Results

To show the performance of our deep learning system in classifying handwritten Phoenician characters, we report in Table 2 and Table 7 (Appendix B) respectively the classification rate of each Phoenician character and the confusion matrix corresponding to the best performance recorded by the system. We observe a relatively high overall classification rate with an accuracy of 0.9851, meaning that our ConvNet is successful at detecting relevant features from handwritten Phoenician characters.
Also, we observe that the number of misclassified characters is very small compared to the well-predicted ones. The misrecognition of the most characters is due to two factors: the presence of badly written characters within the database such as the ones shown in Figure 5, and structural similarities

Table 2: Classification rate of each character in the PHCDB database during the validation phase.

Character	Accuracy	Character	Accuracy
	1		0.9753
	0.9815		0.9938
	0.9630		0.9753
	0.9753		1
	0.9815		0.9877
	1		0.9877
	1		0.9815
	1		0.9321
	0.9877		0.9938
	0.9691		1
	0.9938	**Average**	**0.9851**
	0.9938		

between some Phoenician characters namely  and ,  and ,  and  as illustrated in the confusion matrix (Table 7).

Figure 5: Examples of badly written characters in our database.

On the other hand, to show the effectiveness of our technique over other techniques, we compare it with two different recognition approaches: (i) the Deep Belief Network (DBN) whose detailed configuration and experimental setup are detailed in Appendix A, and (ii) a technique based on HOG (Histogram of Oriented Gradient) features [13] and a multiclass SVM (Support Vector Machine) classifier, where the HOG cell size is set to 4×4. Note that we retain only the best performance for each of these approaches and report them in Table 3. From results of Table 3, we observe that the proposed ConvNet surpasses by far the other approaches.

Table 3: Comparison of different used classifiers.

	HOG-SVM	DBN	Phoenician ConvNet
Accuracy	0.9052	0.8566	0.9851

4 TRANSFER LEARNING FOR HANDWRITTEN TIFINAGH CHARACTER RECOGNITION

The Tifinagh alphabet, also known as Libyan or Berber script, is the writing system of the Amazigh language and is composed of 33 phonetic entities (Figure 7a). Its ancient version existed from the 3rd century BC to the 3rd century AD and was more widely used in North Africa. According to [14], the question of the origin of the Lybian script is still under debate and there are three theories in this regard: i) borrowing and adaptation from another ancient written form especially Phoenician, ii) autochthonous creation from a stock of ancient signs, and iii) a mixture of the two previous theories. In this section, we want to affirm the 1st theory (i.e., Tifinagh is derived from Phoenician) by showing that Tifinagh and Phoenician characters share common features. This is done through: (i) the application of the transfer learning (TL) technique (with full fine-tuning) from the Phoenician to the Tifinagh alphabet whereby features learned from Phoenician character shapes are used to learn Tifinagh characters' shapes, which yields better classification results than the proposed *Phoenician ConvNet* (Section 3), and (ii) the visualization of low-level weights learned from this TL technique compared to those learned from the *Phoenician ConvNet* and from the ConvNet trained on Tifinagh characters.

4.1 Experiment

Dataset. We perform our Tifinagh handwritten character recognition evaluation on the AMHCD database [15]. The dataset has 25740 isolated Amazigh handwritten characters, i.e. 780 characters per class, produced by 60 writers who wrote 13 samples of each class. Characters are further resized proportionally to fit a 60×60 frame.

Simulation details. In this experiment, the ConvNet consists of all layer weights of the pre-trained *Phoenician ConvNet* except last layer ones which are discarded and replaced by randomly initialized ones. Then, this ConvNet is trained on the Tifinagh alphabet dataset. Training hyper-parameters are the same as Section 3, except that the number of epochs needed for full convergence is reduced to 35 epochs. This experiment is referred to as *Full TL using Phoenician ConvNet*.

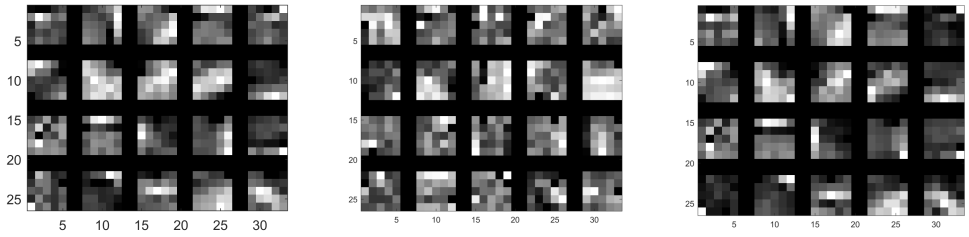
Comparison with other techniques. To illustrate the effectiveness of the proposed approach, we compare our transfer learning approach against existing strategies, namely: (i) the geometrical technique proposed by [16], approaches based on detecting (ii) the horizontal and vertical centerline of characters [17] and (iii) the horizontal and vertical baseline of characters [18], (iv) the work of [19] based on continuous HMMs and directional features, (v) the technique of [20] combining multiple classifiers with statistical features, (vi) our previous work [6] in which we trained a simple ConvNet, and (vi) another work based on training a ConvNet [21]. Another technique to compare with our approach is to train a randomly initialized ConvNet (whose architecture and training setup are the same as in section 3) on the Tifinagh alphabet dataset, which will be referred to as *Tifinagh ConvNet*.

4.2 Results

Results of the proposed *Full TL using Phoenician ConvNet* technique as well as state-of-the-art techniques are reported in Table 4. Results show that our proposed method is successful at classifying Tifinagh characters and even surpasses all existing works with an accuracy of 99.05%. Indeed, it outperforms all hand-crafted feature works as well as automated feature works.

Table 4: Comparison between our method and other existing approaches.

Method used	Total used images from AMHCD (%)	Training set size (percentage wrt. total size)	Test set size	Accuracy
Geometrical method [16]	1700 (06.60%)	1000 (59%)	700	92.30
Horizontal and vertical centerline of character [17]	20,150 (78.28%)	18,135 (90%)	2015	96.32
Horizontal and vertical baseline of character [18]	24,180 (93.93%)	21,762 (90%)	2418	94.96
Continuous HMMs and directional features [22]	20,180 (93.93%)	16120 (80%)	8060	97.89
Combining multiple classifiers with statistical features [23]	24,180 (93.93%)	16,926 (70%)	7254	99.03
ConvNet of [6]	24,180 (93.93%)	16,120 (67%)	8060	98.25
ConvNet of [21]	25,740 (100%)	18,018 (70%)	7722	99.00
Tifinagh ConvNet	25,740 (100%)	17,160 (67%)	8580	98.95
Proposed transfer learning method	25,740 (100%)	17,160 (67%)	8580	99.05



(a) (b) (c)

Figure 6: The 5 by 5 filter weights of the 1st convolutional layer for the following approaches: (a) Phoenician ConvNet, (b) Tifinagh ConvNet, (c) Full TL using Phoenician ConvNet.

Moreover, comparing results of our TL technique and the *Tifinagh ConvNet* shows that training the ConvNet with weights learned from Phoenician alphabets yields better performance than training the ConvNet with randomly initialized weights. This suggests that features learned from Phoenician are efficient in the transfer learning task in hand and that Phoenician and Tifinagh characters may share common features. To illustrate this fact, we compare 1st convolutional layer filters of the *Tifinagh ConvNet*, filters of the *Full TL using Phoenician ConvNet* and filters of the *Phoenician ConvNet* by plotting their corresponding weight matrices as illustrated in Figure 6. By analyzing the type of visual features learned by neurons of the 1st convolutional layers, the following observations are made:

(i) 1st layer filters of both *Phoenician ConvNet* and *Full TL using Phoenician ConvNet* are clearer and present stronger edges (i.e., horizontal and vertical fluctuations) than *Tifinagh ConvNet* weights, suggesting that Phoenician character shapes help in capturing more low-level features which are relevant enough for the Tifinagh character recognition task;

(ii) *Phoenician ConvNet* and *Full TL using Phoenician ConvNet* 1st layer weights are almost the same with a light modification on some of the filters. So, even after *Phoenician ConvNet* weights have been updated and fine-tuned on Tifinagh characters, they remained unchanged, suggesting that Tifinagh and Phoenician characters share common low-level features or elementary shapes.

5. TRANSFER LEARNING FOR RECOGNIZING HANDWRITTEN ALPHABETS WITH LACK OF ANNOTATED DATA

5.1 Methodology

Since most existing alphabets are derived from the Phoenician alphabet, this latter must probably share common features with these alphabets and could therefore be employed to improve the recognition of existing alphabet characters. These features can be useful especially in the case when there is a lack of annotated data within a given alphabet.

As we know, in order for a ConvNet to be efficient in feature extraction and classification, it needs to be trained by a relatively large number of training instances for full convergence. And, one way to solve the issue of few labeled data within a target domain is to first train the ConvNet on

data belonging to a domain similar to the target data domain and then fine-tune it with target data. To this matter, we propose a light-weight and global transfer learning system which has the following properties: (i) it freezes all weights of the *Phoenician ConvNet* (of Section 3) except the last layer weights and replaces these last layer weights by new randomly initialized weights (the number of weights being set according to the number of characters within the target alphabet), (ii) it fine-tunes last layer weights by training the new ConvNet on the target alphabet characters. As such, the number of parameters to be learned within the system is limited, which makes the training process much faster.

5.2 Datasets

To determine how well the proposed transfer learning system works, we conduct a series of experiments on a number of different target alphabet datasets for character classification, namely Tifinagh, Latin, Arabic, Russian and Bengali handwriting character datasets.

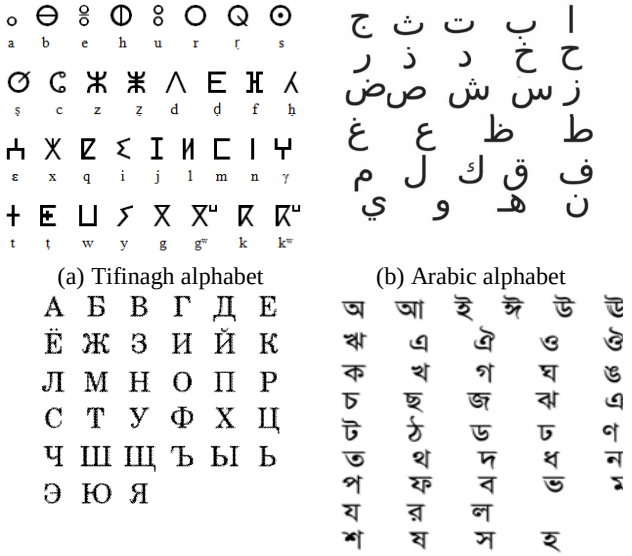
Latin dataset. The Latin dataset is taken from the Emnist database [24]. This repository contains 12,051 images (of size 28×28) of the 31 Latin characters.

Arabic dataset. The Arabic script is written from right to left and is composed of 28 characters (Figure 7.a). Arabic characters data is taken from the Handwritten Arabic Characters Database (HACDB) [25] which has two versions: one with 66 shapes or labels (58 shape characters and 8 shapes of overlapping characters) and a basic one with 24 shapes (representing 24 basic characters). The latter version is the one used in our study and is composed of a total of 2400 jpg images (of size 128×128).

Russian dataset. The Russian alphabet uses letters from the Cyrillic script to write the Russian language (Figure 7.c). Russian characters data is taken from the Cyrillic-oriented MNIST dataset (CoMNIST) [26] which is a free, crowd-sourced version of MNIST that contains digitalized letters from the Cyrillic and Latin alphabet. The Cyrillic repository currently contains 15,233 png handwritten images of size 278×278 representing all 33 letters of the Russian alphabet.

Bengali dataset. Bengali is the native language of Bangladesh (Figure 7.d). The Bengali character dataset is obtained from the Handwritten Indian script character database CMATERdb 3 [27]. The dataset contains 15000 images representing the 50 basic Bengali characters.

Characters of these target datasets are further resized proportionally to fit a 60 × 60 frame.



(a) Tifinagh alphabet (b) Arabic alphabet

(c) Russian alphabet (d) Bengali alphabet

Figure 7: Some existing alphabets.

5.3 Experiments and results

Experiment 1. The Phoenician transfer learning system consists of retraining the *Phoenician ConvNet* according to steps of the methodology (section 5.1) for each and every target dataset (mentioned in Section 5.2). The ConvNet training hyper-parameters are the same as in the section 3, except that the number of epochs needed for full convergence is reduced to 35 epochs. Furthermore, training is conducted using only a subset of the target datasets (i.e., n training instances). This experiment is referred to as *TL using Phoenician ConvNet*.

In order to prove the effectiveness of our *TL using Phoenician ConvNet* technique, we compare it against the following transfer learning (TL) systems:

(i) *TL using Latin ConvNet*: we first train a ConvNet on the whole Latin dataset using the same network architecture, simulation settings and cross validation as in section 3. Then, we fine-tune it (according to methodology of section 5.1) for 35 epochs on n instances of each of the target datasets (except the Latin dataset);

(ii) *TL using Digits ConvNet:* we first train a ConvNet on digit images contained within the training set of the MNIST dataset [28], based on the same network architecture and simulation settings as in section 3. Digit images are resized from 28×28 to 60×60 before being fed to the network. Then, we fine-tune this ConvNet (according to methodology of section 5.1) for 35 epochs on n instances of each of the target datasets;

(iii) *TL using Cifar ConvNet:* we train a ConvNet on natural images contained within the training set of the Cifar dataset [29], based on the same network architecture and simulation settings as in section 3. Digit images are resized from 28×28 to 60×60 before being fed to the

network. Then, we fine-tune this ConvNet (according to methodology of section 5.1) using the same n instances of each of the target datasets;

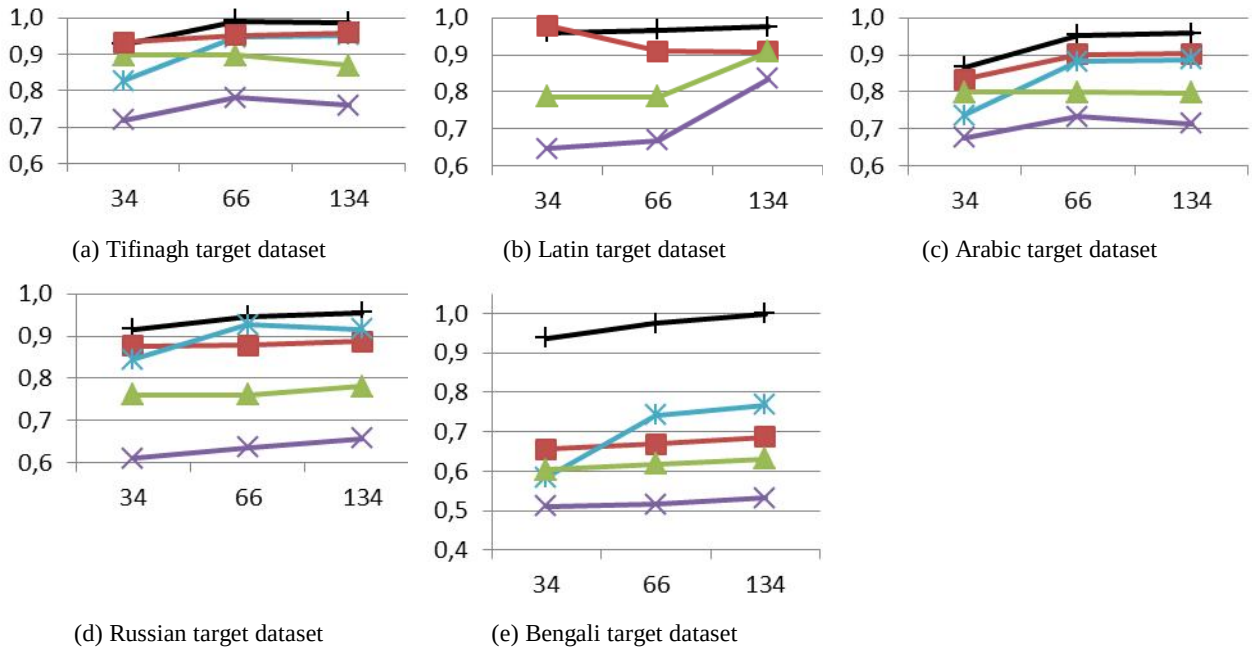
(iv) *RI ConvNet:* we train one randomly initialized (RI) ConvNet for 70 epochs on n instances of each of the mentioned target datasets, using the same network architecture, simulation settings and cross validation as in Section 3.

For this experiment, the number of training instances n is set to 34 instances.

Corresponding results are reported in Table 5. From these results, we observe that all TL systems are able to recognize target datasets but each at different rates. Indeed, *TL using Phoenician ConvNet* technique surpasses *TL using Latin ConvNet*, *TL using Digits ConvNet* and *TL using Cifar ConvNet* techniques on all target alphabets, implying that pre-training the ConvNet on Phoenician characters and retraining its last layer only on one of the target datasets yields a better recognition than pre-training it on Latin letters, digits or natural Cifar images. So, features learned from Phoenician

Table 5: Comparative results using different Transfer Learning ConvNets.

Target dataset	RI ConvNet	TL using Phoenician ConvNet	TL using Latin ConvNet	TL using Digits ConvNet	TL using Cifar ConvNet
Tifinagh	0.9277	**0.9327**	0.8266	0.8970	0.7184
Latin	0.9594	**0.9786**	_	0.7868	0.6471
Arabic	**0.8657**	0.8333	0.7361	0.7980	0.6765
Russian	**0.9158**	0.8771	0.8451	0.7610	0.6114
Bengali	**0.8867**	0.6056	0.5333	0.5533	0.4612



(a) Tifinagh target dataset (b) Latin target dataset (c) Arabic target dataset

(d) Russian target dataset (e) Bengali target dataset

Figure 8: Classification results of the *RI ConvNet* (black), *TL using Phoenician ConvNe* (red), *TL using Latin ConvNet* (blue), *TL using Digits ConvNet* (green), and *TL using Cifar ConvNet* (purple) techniques as a function of the number of training instances n per target dataset.

characters generalize better on existing alphabets than features learned from Latin characters, digits or natural images, which suggests that the features of the Phoenician alphabet are more global than features of other alphabets. Also, the *Phoenician ConvNet* can serve as a baseline ConvNet to train any alphabet, and the *TL using Phoenician ConvNet* system can be regarded as an efficient technique to recognize any alphabet with few labeled data.

Comparing the *TL using Phoenician ConvNet* technique with *RI ConvNet* technique shows that the former surpasses the latter when dealing with Tifinagh and Latin as target datasets and is less than the latter for Arabic, Russian and Bengali target datasets. However, let's note that, according to Table 8, *RI ConvNet* requires training a total of 3,248,892 parameters for 70 epochs as opposed to *TL using Phoenician ConvNet* system which requires training 11,022 parameters only for 35 epochs. Therefore, *TL using Phoenician ConvNet* can be regarded as a light-weight system for the recognition of any alphabet with few labeled data.

Experiment 2. We run the same simulations as in Experiment 1 with $n = 66$ and $n = 134$. Given the set of training instances $n = \{34, 66, 134\}$, classification results of these techniques on the Tifinagh, Latin, Arabic, Russian, and Bengali target datasets are displayed in Figures 8.a, 8.b, 8.c, 8.d and 8.e respectively. From these results, the following observations can be made:

(i) As the number of training instances n increases, all techniques have a higher classification rate on most of the target datasets;

(ii) Moreover, increasing n makes the recognition performance of *TL using Phoenician ConvNet* on all target datasets comparable to that of *TL using Latin ConvNet*, meaning that the ConvNet pre-trained on Latin characters may hold features as global as the Phoenician ConvNet but that the last layer (representing the classifier) of the former ConvNet needs more training instances to converge.

(iii) Even after increasing the number of training instances, we still have the *RI ConvNet* the best at recognizing alphabets, followed by *TL using Phoenician ConvNet* and *TL using Latin ConvNet*, then by *TL using Digits ConvNet*, then by *TL using Cifar ConvNet*. However, as mentioned in the previous experiment, *RI ConvNet* remains a computationally heavy technique compared with the rest of the techniques, making the *TL using Phoenician ConvNet* the most optimal technique for recognizing alphabets with few labeled data.

6. CONCLUSION

In order to promote the use of standardized data sets and evaluation methods for research in matching and recognition of old Phoenician scripts, a reference database of Phoenician handwriting is needed. As such, we proposed a Phoenician Handwritten dataset (PHCDB) which provides a repository of handwritten Phoenician characters containing 10,714 character shapes. Then, we further used this database to train and test a deep learning system (a Convolutional Neural Network) for the recognition of handwritten Phoenician characters. Afterwards, we introduced a transfer learning system with Phoenician characters as the source domain in order to improve the recognition of handwritten Tifinagh characters. Thanks to this system, we proved that there exists a correspondence between the Tifinagh and Phoenician scripts and that the former could have been derived from the latter. Finally, we developed a fast, global and light-weight transfer learning network based on Phoenician character shapes which can be used for the recognition of any alphabet which experiences a lack of annotated data.

7. DATA AVAILABILITY

The data that support the findings of this study are openly available in the repository 'Phoenician_recognition_code' at the URL:
https://github.com/lsadouk/Phoenician_recognition_code

8. APPENDIX

The configuration settings and architecture of the Deep Belief Network (section 3) are detailed in Appendix A. A summary of the *Phoenician ConvNet* architecture and parameters (section 3) as well as the confusion matrix for the Phoenician ConvNet (section 3) are illustrated in Appendix B.

8.1 Appendix A

Table 6: Learning parameters of the DBN set according to suggestions published by Hinton [30].

	Learning rate	momentum	dropout	# epochs	Batch size
Unsupervised phase	0.0001	0.7	0.3	200	200
Supervised phase	0.0001	0.9	_	100	200

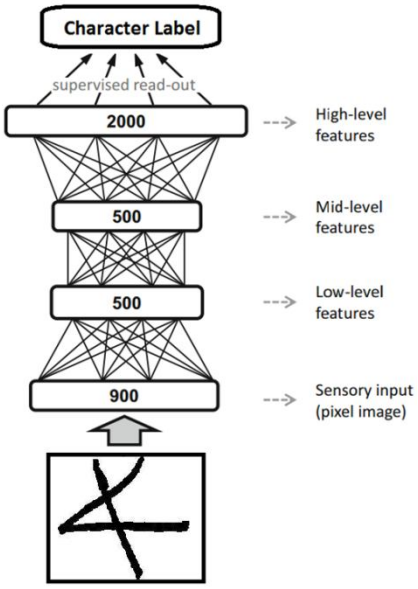


Figure 9: Graphical representation of our DBN model: three RBMs with 500 hidden neurons in the first and second layers, and 2000 hidden neurons in the third layer. Undirected connections account for unsupervised, generative learning, while directed ones entail supervised learning.

8.2 Appendix B

Table 7. Confusion matrix for the *Phoenician ConvNet*. Rows and columns represent the actual and predicted values respectively.

	𐤀	𐤁	𐤂	𐤃	𐤄	𐤅	𐤆	𐤇	𐤈	𐤉	𐤊	𐤋	𐤌	𐤍	𐤎	𐤏	𐤐	𐤑	𐤒	𐤓	𐤔	𐤕
𐤀	162																					
𐤁		159									1									2		
𐤂		1	156														5					
𐤃				158				1				1				1				1		
𐤄		1			159						1				1							
𐤅						162																
𐤆							162															
𐤇								162														
𐤈									160							1		1				
𐤉		1			2				1	157					1							
𐤊											161										1	
𐤋	1											161										
𐤌													158	3								1
𐤍													1	161								
𐤎	1						1				1				158				1			
𐤏																162						
𐤐			1											1			160					
𐤑																		160	1		1	
𐤒									2							1			159			
𐤓		8	2	1																151		
𐤔													1								161	
𐤕																						162

Table 8. A summary of the ConvNet architecture parameters.

Id	Layer	Type	Filter size	Filter Num	Stride	Pad	Output Size	#Parameters
0		Input	-				60 × 60	0
1	L1	Conv	5 × 5	20	1	0	56 × 56	520
2		ReLu	-	-	-	-	56 × 56	0
3		Maxpool	2 × 2	-	2	0	28 × 28	0
4	L2	Conv	4 × 4	50	1	0	25 × 25	16,050
5		ReLu	-	-	-	-	25 × 25	0
6		Maxpool	2 × 2	-	2	0	12 × 12	0
7	L3	Conv	3 × 3	150	1	1	12 × 12	67,650
8		ReLu	-	-	-	-	12 × 12	0
9	L4	Conv	3 × 3	150	1	1	12 × 12	202,650
10		ReLu	-	-	-	-	12 × 12	0
11		Maxpool	2 × 2	-	2	0	6 × 6	0
12	L5	Conv (FC1)	6 × 6	500	1	0	1 × 1	2,700,500
13		ReLu	-	-	-	-	1 × 1	0
14		Dropout	-	-	-	-	1 × 1	0
15	L6	Conv (FC2)	1 × 1	500	1	0	1 × 1	250,500
16		ReLu	-	-	-	-	1 × 1	0
17		Dropout	-	-	-	-	1 × 1	0
18	L7	Conv (FC3)	1 × 1	33	1	0	1 × 1	11,022

REFERENCES

1. P. Xella and J. Á. Zamora, **7 Phoenician Digital Epigraphy: CIP Project, the State of the Art,** in Crossing Experiences in Digital Epigraphy, 2019, pp. 93–101. https://doi.org/10.1515/9783110607208-008
2. J. Cunchillos, P. Xella, and J. Zamora, **Il corpus informatizzato delle iscrizioni fenicie e puniche: un progetto italo-spagnolo**, Univ. di Palermo, 2005.
3. A. M. Alqudah, H. Alquraan, I. Abu Qasmieh, A. Alqudah and W. Al-Sharu, **Brain Tumor Classification Using Deep Learning Technique - A Comparison between Cropped, Uncropped, and Segmented Lesion Images with Different Sizes**, Volume 8, No.6, pp. 3684 – 3691, 2019. https://doi.org/10.30534/ijatcse/2019/155862019

4. M. Akour, O. Al Qasem, H. Alsghaier and K. Al-Radaideh, **The Effectiveness of Using Deep Learning Algorithms in Predicting Daily Activities**, Volume 8, No. 5, pp. 2231 - 2235, 2019. https://doi.org/10.30534/ijatcse/2019/57852019
5. J. R. B. Del Rosario, **Development of a Face Recognition System Using Deep Convolutional Neural Network in a Multi-view Vision Environment**, Volume 8, No.3, pp. 369 – 374, 2019. https://doi.org/10.30534/ijatcse/2019/06832019
6. 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*, 2017, p. 59. https://doi.org/10.1145/3109761.3109788
7. D. Soulaimani, **Writing and rewriting Amazigh/Berber identity: Orthographies and language ideologies**, *Writ. Syst. Res.*, vol. 8, no. 1, pp. 1–16, Jan. 2016.
8. J. J. Mark, **Phoenicia**, *Ancient History Encyclopedia*. 2018.
9. J. Kaltner and S. L. McKenzie, **Beyond Babel: a handbook for biblical Hebrew and related languages**, vol. 42. 2019. https://doi.org/10.2307/j.ctvbkk0x6
10. M. Howard, **Transnationalism in ancient and medieval societies: The role of cross-border trade and travel**. 2014.
11. S. Fischer, **History of Writing**, Reaktion b. 2003.
12. I. Goodfellow, Y. Bengio, and A. Courville, **Deep learning**. *MIT Press*, 2016.
13. N. Dalal and W. Triggs, **Histograms of Oriented Gradients for Human Detection**, in *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. CVPR05*, vol. 1, no. 4, pp. 886–893, 2005.
14. J.-L. Le Quellec, **Rock art, scripts and proto-scripts in Africa: The Libyco-Berber example**, *Writ. Cult. a Colon. Context Africa Am.*, pp. 1500–1900, 2011.
15. Y. Es Saady, A. Rachidi, M. El Yassa, and D. Mammass, **AMHCD : A Database for Amazigh Handwritten Character Recognition Research**, *Int. J. Comput. Appl.*, vol. 27, no. 4, pp. 44–48, 2011. https://doi.org/10.5120/3286-4475
16. 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*, 1997, vol. 2, pp. 564–567 vol.2.
17. Y. Es Saady, A. Rachidi, M. El Yassa, and D. Mammass, **Amazigh Handwritten Character Recognition based on Horizontal and Vertical Centerline of Character**, *Int. J. Adv. Sci. Technol.*, vol. 33, pp. 33–50, 2011.
18. Y. Es Saady, M. Amrouch, A. Rachidi, M. El Yassa, and D. Mammass, **Handwritten Tifinagh Character Recognition Using Baselines Detection Features**, *Int. J. Sci. Eng. Res.*, vol. 5, no. 4, pp. 1177–1182, 2014.
19. M. Amrouch, Y. Es-Saady, … A. R.-… J. of M., and undefined 2012, **Handwritten amazigh character recognition system based on continuous hmms and directional features**, pdfs.semanticscholar.org.
20. N. Aharrane, A. Dahmouni, K. El Moutaouakil, and K. Satori, **A robust statistical set of features for Amazigh handwritten characters**, *Pattern Recognit. Image Anal.*, vol. 27, no. 1, pp. 41–52, Jan. 2017.
21. M. Benaddy, O. El Meslouhi, Y. Es-saady, and M. Kardouchi, **Handwritten Tifinagh Characters Recognition Using Deep Convolutional Neural Networks**, *Sens. Imaging*, vol. 20, no. 1, p. 9, Dec. 2019. https://doi.org/10.1007/s11220-019-0231-5
22. M. Amrouch, A. Rachidi, M. El Yassa, and D. Mammass, **Handwritten Amazigh Character Recognition System Based on Continuous HMMs and Directional Features**, *Int. J. Mod. Eng. Res.*, vol. 2, no. 2, pp. 436–441, 2012.
23. N. Aharrane, A. Dahmouni, K. El Moutaouakil, and K. Satori, **A robust statistical set of features for Amazigh handwritten characters**, *Pattern Recognit. Image Anal.*, vol. 27, no. 1, pp. 41–52, Jan. 2017.
24. G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, **EMNIST: Extending MNIST to handwritten letters**, in *Proceedings of the International Joint Conference on Neural Networks*, 2017, vol. 2017–May, pp. 2921–2926.
25. A. Lawgali, M. Angelova, and A. Bouridane, **HACDB: Handwritten Arabic Characters Database for Automatic Character Recognition**, in *Visual Information Processing (EUVIP), 2013 4th European Workshop on*, 2013, pp. 255–259.
26. Comnist, **Cyrillic oriented MNIST: A dataset of Latin and Cyrillic letter images**, *Website of Kaggle Inc.*, 2019. [Online]. Available: https://www.kaggle.com/gregvial/comnist.
27. N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, **A genetic algorithm based region sampling for selection of local features in**

handwritten digit recognition application, *Appl. Soft Comput. J.*, vol. 12, no. 5, pp. 1592–1606, 2012. https://doi.org/10.1016/j.asoc.2011.11.030

28. Li Deng, **The MNIST Database of Handwritten Digit Images for Machine Learning Research**, *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 141–142, 2012. https://doi.org/10.1109/MSP.2012.2211477
29. A. Krizhevsky and G. Hinton, **Learning multiple layers of features from tiny images**, 2009.
30. G. E. Hinton, S. Osindero, and Y.-W. Teh, **A Fast Learning Algorithm for Deep Belief Nets**, *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006. https://doi.org/10.1162/neco.2006.18.7.1527