International Journal of Computer Applications (0975 - 8887)
Volume 176 - No.4, October 2017

Deep Arabic Font Family and Font Size Recognition

Ibrahim M. Amer
Computer Science Department,
Faculty of Computer &
Information Sciences,
Ain Shams University,
Cairo 11566, Egypt

Salma Hamdy
Computer Science Department,
Faculty of Computer &
Information Sciences,
Ain Shams University,
Cairo 11566, Egypt

Mostafa G. M. Mostafa
Computer Science Department,
Faculty of Computer &
Information Sciences,
Ain Shams University,
Cairo 11566, Egypt

ABSTRACT

Font family and font size recognition became an essential step for document analysis. Font recognition helps to identify the proper segmentation method to be used before feeding the document to the Optical Character Recognition (OCR). In this paper, some of the previous techniques used for font family and font size recognition will be discussed then we will discuss the proposed method that is based on deep learning. Two methods have been presented in this paper 1) a method for font family recognition (font size invariant) and 2) a method for font size recognition. Both methods use Deep Convolutional Neural Networks (D-CNN). We evaluated the proposed method on Arabic Printed Text Image Database (APTI) [7] and on a document generated using APTI database word images and scanned with the scanner.

Keywords

Font Family Recognition, Font Size Recognition, Optical Character Recognition (OCR), Document Layout Analysis (DLA), Deep Learning, Deep Convolutional Neural Network (D-CNN)

1. INTRODUCTION

Accurate font recognition (font family and font size) is a very important step for document layout analysis and OCR systems, it helps these systems to classify the type of the font being processed so it can be segmented and classified using the proper segmentation and learning method. Research on font recognition has got a lot of focus recently as it is very important for OCR systems. The cursive nature of Arabic words makes developing an Arabic OCR much more complex than OCRs for other languages. A segmentation algorithm tailored for a specific Arabic font is not guaranteed to work on another font so, it is important to develop (in some cases) a segmentation algorithm for each font family to achieve good OCR results. So, font recognition is useful for improving text recognition in terms of accuracy and time [1].

Fouad Slimane et. al [2] proposed a method for font family/size recognition using Gaussian Mixture Models (GMM). This method treats a word image with a fixed-length, overlapping sliding window. Each window has 102 features captured by Gaussian Mixture Models. This paper presents three systems: a font recognition system, a size recognition system and a font and size recognition system.

All word images are grayscale. Some of the features extracted from the grayscale images and some others from binary images. All word images were scaled to 30 pixels height and transformed into a sequence of feature vectors computed using a four pixels sliding window over the word image, no segmentation into letters is applied and the whole word image is transformed into feature vectors. There are two parts, the first part extracts for each window 12 different features presented in the paper that are computed using the gray level values of the images. The second part of the feature extraction consists of resizing the window to 10 pixels height and then computing both horizontal and vertical projection histogram values. The overall feature vector consists of 51 coefficients, but after computing the delta between each two consecutive vectors, the size of the vector becomes 102. For font recognition the results where 99.6% on APTI dataset. Size recognition results varied from 92.0% to 99.3% for different font families.

Faten Kallel Jaiem, Slim Kanoun, Veronique Eglin [3] used Steerable Pyramid (SP) for texture analysis of Arabic homogeneous and normalized text block for font recognition. The classification is done using KNN and Backpropagation algorithm on APTID/MF database. The steerable pyramid is used to extract texture features from text image. Steerable pyramid is a bank of filters applied to the image in 6 different orientations 0, 30,60, 90, 120 and 150. Initially, the image is separated into low and high-pass sub-bands, using filters L0 and H0. The low-pass sub-band is then divided into a set of oriented band-pass sub-bands and a low(er)-pass sub-band. This low(er)-pass sub-band is sub-sampled by a factor of 2 in the X and Y directions. The recursive (pyramid) construction of a pyramid is achieved by inserting a copy of the shaded portion of the diagram at the location of the solid circle (i.e., the low-pass branch). The experimental results showed that the use of steerable pyramid with 6 orientations give high recognition rates about 99%.

Mahmoud A. A. Mousa, Mohammed S. Sayed,& Mahmoud I. Abdalla [4] introduces an algorithm for Arabic font recognition for 10 font types. The algorithm uses scale-invariant key points detectors to match font types. Many key points detectors have been used, such as SIFT, DoG, Hessian, Hessian Laplace, Harris, D-sift and so on after that, the classification is done using K-means clustering algorithm.The proposed algorithm produces a mean recognition rate of 99.2 - 99.5%.

Alican Bozkurt, Pinar Duygulu & A. Enis Cetin [5] proposed a method for recognizing fonts. It addressed problems as analysis and categorization of textures using complex wavelet transform and Support Vector Machine (SVM). The paper has been evaluated on different datasets in four languages. The proposed method was as follows: 1. Pre-processing: the proposed method works on multi-font documents. The method is capable of detecting empty areas as follows: the binarized document (Otsu's Method) is divided into blocks, the block is marked empty if the ratio of black pixels to white pixels is below a certain threshold. Then CWT (Complex wavelet transform) is used because it is more reliable than the ordinary Discrete wavelet transform. The feature vector includes mean and variance values of 18 output images (six outputs per level of a three-level complex wavelet tree), resulting in a 36-element feature vector. They used SVM for classification with the radial basis function (RBF) for the kernel function. (LIBSVM). The average recognition accuracy for Arabic fonts was 96.86%.

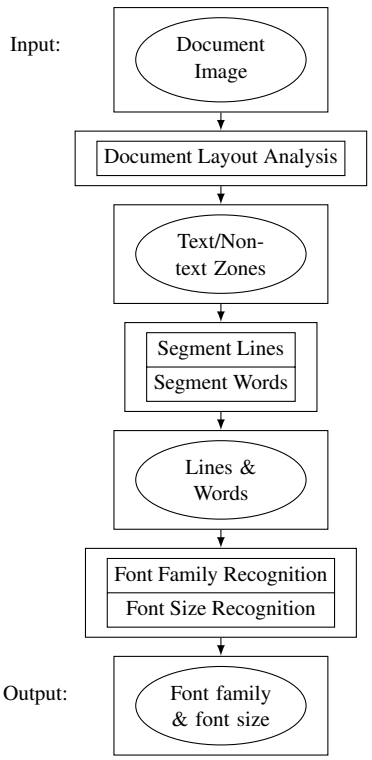

Fig. 1: Proposed System Architecture

2. PROPOSED METHOD

In [6] we proposed a method for Arabic document layout analysis for text localization and separation from images in Arabic newspapers. The method mainly uses a deep convolutional neural network (D-CNN) to classify zones/patches of the document image as text or non-text then the text lines and words of textual zones are extracted. As mentioned before, font family and font size recognition are important steps for document layout analysis and can significantly enhance document layout analysis process. In this section, the proposed method for both font family and font size recognition will be described in details. The system architecture is shown in figure 1.

2.1 Font Family Recognition

For font family recognition, we used a D-CNN to classify ten different font families (fonts supported by APTI dataset). The proposed method is a multi font-size recognition method; which means that the method is size invariant which can recognize fonts with ten different font sizes (6, 7, 8, 9, 10, 12, 14, 16, 18 and 24). Four different D-CNNs were trained according to the following:

- A D-CNN to classify two font families only (Andalus and DecoTypeNaskh) with one font size.
- A D-CNN to classify four font families (Andalus, DecoTypNaskh, SimplifiedArabic and Tahoma) with one font size.
- A D-CNN to classify all available font families (Advertising Bold, Andalus, Arabic Transparent, Deco Type Naskh, Deco Type Thuluth, Diwani Letter, MUnicode Sara, Simplified Arabic, Tahoma and Traditional Arabic) with one font size.
- A D-CNN to classify all font families with ten different font sizes (all font families and sizes supported by APTI).

The following sections will discuss more about the architectures used.

2.1.1 Pre-processing.

All training and testing images have been re-scaled to 40×40 pixels. Two feature scaling methods have been used. For the first three networks, each image was divided by the maximum intensity value (255) over the whole training and testing set. For the last network, mean centering or mean normalization was used so that the data will have zero mean as in equation 1.

$$X = (X - \mu)/S \tag{1}$$

Where X is the target feature, μ is the feature's mean and S is the feature's range of values (in our case $S = 255$ which is the maximum intensity value of the pixel).

2.1.2 Architecture.

Four deep convolutional neural networks were trained to classify different font families according to the following architectures:

The first trained network architecture was as follows:
Two convolutional layers with 32 feature maps, 3×3 kernel size and dropout = 0.2 [8]; followed by 2×2 max pooling layer; followed by two convolutional layers with 64 feature maps, 3×3 kernel size and dropout = 0.2; followed by 2×2 max pooling layer; followed by two convolutional layers with 128 feature maps, 3×3 kernel size and dropout = 0.2; followed by 2×2 max pooling layer; followed by a fully connected layer of 1024 neurons; followed by a fully connected layer of 512 neurons. The output layer consists of two nodes because we are classifying between two font families. Number of epochs = 25.

The second trained network architecture was as follows: Two convolutional layers with 40 feature maps, 3×3 kernel size and dropout = 0.2; followed by a 2×2 max pooling layer; followed by two convolutional layers of 40 feature maps, 3×3 kernel

size and dropout = 0.2; followed by a 2×2 max pooling layer; followed by Two convolutional layers of 64 feature maps, 3×3 kernel size and dropout = 0.2 followed by a max pooling layer with 2×2 kernel size; followed by two convolutional layers of 128 feature maps, 3×3 kernel size and dropout = 0.2; followed by 2×2 max pooling layer.Finaly, a fully connected layer of 1024 neurons followed by another fully connected layer of 512 neurons followed by an output layer of four classes. Number of epochs = 35.

The third trained network architecture was as follows: Two convolutional layers with 40 feature maps, 3×3 kernel size and dropout = 0.2; 2×2 max pooling layer; followed by two convolutional layers of 40 feature maps, 3×3 kernel size and dropout = 0.2; 2×2 max pooling layer; followed by two convolutional layers of 40 feature maps, 3×3 kernel size and dropout = 0.2; 2×2 max pooling layer; followed by two convolutional layers of 64 feature maps, 3×3 kernel size and dropout = 0.2; 2×2 max pooling layer; followed by two convolutional layers of 128 feature maps, 3×3 kernel size and dropout = 0.2; 2×2 max pooling layer. Finally a fully connected layer of 1024 neurons followed by another fully connected layer of 512 neurons and an output layer of 10 nodes to classify ten different font families.Number of epochs = 35.

The fourth and the last trained network architecture was as follows: Two consecutive convolutional layers of 64 feature maps with 5×5 kernel sizes respectively and dropout = 0.2; followed by two convolutional layers of 128 feature maps with 3×3 kernel size and a dropout = 0.2;a max pooling layer of 2×2; followed by a convolutional layers of 256 feature maps with 3×3 kernel size. Finally a fully connected layer of 1024 neurons followed by another fully connected layer of 512 neurons and an output layer (softmax layer) of ten nodes.Number of epochs = 50.

For all trained networks we used Adaptive Moment Estimation (Adam)[9] optimization algorithm and Rectified Linear Unit (RELU)[10] as an activation function. Figures 4 and 5 show a demonstration for the fourth architecture used and a visualization for some convolutional filters used in the network respectively.

2.2 Font Size Recognition

For font size recognition, we used deep convolutional neural network as well to classify font sizes. Given the font family which is previously recognized by the font recognition model, **ten D-CNNs** were trained to classify ten font sizes (6, 7, 8, 9, 10, 12, 14, 16, 18 and 24) for each font family. The architecture for all models is the same:

2.2.1 Pre-processing and feature extraction.

APTI dataset was used as well for training and testing. All training and testing images have been pre-processed to extract features from them. For each image, both horizontal (HPP) and vertical projection (VPP) profiles are computed and concatenated (equation 2) then combined with the height of the image to form the feature vector. The resultant feature vector size differs according to the image width and height so, all the feature vectors must be of the same size to fed them to the classifier; thus, all feature vectors have been reshaped to the maximum feature vector size found across all training and testing data by appending zeros to each feature vector smaller than the maximum.

$$VPP(y) = \sum_{0<x<w} f(x,y); HPP(x) = \sum_{0<y<h} f(x,y) \quad (2)$$

Where $f(x,y)$ is the input image, w is the image width, h is the image height, $VPP(y)$ is the sum of each row of the image and $HPP(x)$ is the sum of each column of the image.

2.2.2 Architecture.

The architecture used was as follows:
Two convolutional layers of 64 feature maps with 3×3 kernel size each; followed by another convolutional layer of 128 feature maps, 2×2 kernel size and dropout = 0.5. Finally, three consecutive fully connected layers of 1024, 512 and 512 neurons respectively followed by a softmax layer of ten nodes to classify ten output classes (ten font sizes).

3. RESULTS AND DISCUSSION

3.1 Dataset

3.1.1 Arabic Printed Text Image database.

APTI [7] is a dataset for Arabic printed words. It contains 113,284 word images of several font families, styles and sizes. The database is used for competitions related to research of Arabic OCR. The database contains ten different font families with four font styles (Italic, Bold, Plain and Bold Italic) and with ten font sizes. APTI fonts are shown in Fig. 2.

A : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
B : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
C : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
D : نقدم في هذا البحث قاعدة بيانات لكلمات عربية
E : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
F : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
G : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
H : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
I : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية
J : نقدّم في هذا البحث قاعدة بيانات لكلمات عربية

Fig. 2: Fonts used to generate the APTI Database: (A) Andalus, (B) Arabic Transparent, (C) Advertising Bold, (D) Diwani Letter, (E) DecoType Thuluth, (F) Simplified Arabic, (G) Tahoma, (H) Traditional Aatbic, (I) DecoType Naskh, (J) M Unicode Sara.

3.2 Experimental Results

3.2.1 Font family recognition results.

Best accuracy achieved was 97.54% using the 4th trained network (ten fonts / ten font sizes). To improve the accuracy, we replaced the softmax layer with a SVM with RBF kernel. The accuracy improved to 98.6%.

Table 1. : Summary of the results of the four networks (number of training and testing data / accuracy)

Network	#Training / Testing Data	Accuracy
1st Network (two fonts / one size)	65577 / 10000	100.00%
2nd Network (four fonts / one size)	65587 / 10000	99.9%
3rd Network (ten fonts / one size)	168835 / 20000	93.98%
4th Network - CNN + Softmax (ten fonts / ten sizes)	320000/ 80000	97.54%
4th Network - CNN + SVM (ten fonts / ten sizes)	320000/ 80000	98.6%

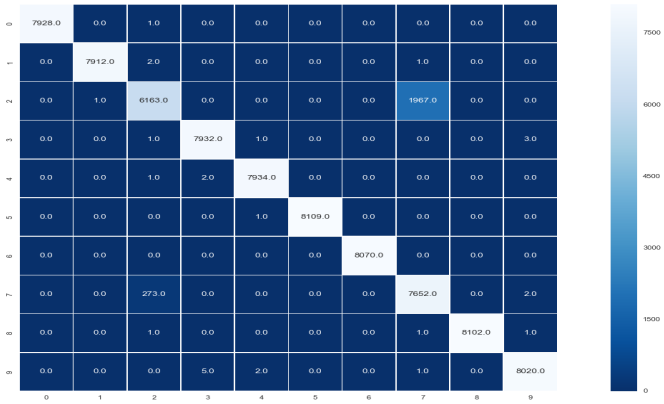

Fig. 3: Confusion Matrix of the 4th network (CNN + SVM for all fonts' families and sizes). Labels shown on x and y axes from 0 to 9 represent font families Advertising Bold, Andalus, Arabic Transparent, DecoType Naskh, DecoType Thuluth, Diwani Letter, M Unicode Sara, Simplified Arabic, Tahoma and Traditional Arabic respectively.

3.2.2 Font size recognition results.

As mentioned before, we trained ten D-CNNs. Each network is trained to recognize the font size of a specific font family.

Table 2. : Summary of the number of training and testing samples used for font size recognition (a D-CNN for each font family)

Font Family	#Training / Testing Data
Advertising Bold	151085 / 37751
Andalus	151085 / 37751
Arabic Transparent	135967 / 33994
DecoType Naskh	151085 / 37751
DecoType Thuluth	151085 / 37751
Diwani Letter	151085 / 37751
M Unicode Sara	151085 / 37751
SimplifiedArabic	151085 / 37751
Tahoma	151085 / 37751
TraditionalArabic	151085 / 37751

The proposed method for font size recognition outperformed the method proposed by Slimane et. al [2] as in table 3.

Table 3. : Comparison between the proposed method and the one proposed by Slimane et. al [2]

Font Family	Slimane et. al	The proposed method
Advertising Bold	96.9%	99.2%
Andalus	99.3 %	99.17%
Arabic Transparent	98.2%	98.94%
DecoType Naskh	92.2%	95.39%
DecoType Thuluth	92.0%	96.12%
Diwani Letter	91.7%	95.77%
M Unicode Sara	98.9%	99.88%
SimplifiedArabic	97.8%	99.34%
Tahoma	98.8%	99.62%
TraditionalArabic	96.2%	96.47%
Average Recognition Accuracy	96.2%	97.99%

For all font size D-CNNs and the fourth font family D-CNN (ten fonts & ten font sizes - CNN + SVM), 10,000 training samples from the total training samples are acquired from word images scanned using a 300 Dots Per Inch (DPI) scanner.

3.3 Testing both font family and size recognition

A document of ten pages has been generated with all fonts supported by APTI dataset. Each page has ten lines, each line represents a font family from APTI, each line of each page represents a font size from APTI and each line has 16 words. The document printed and scanned with 300 DPI. Then the words of each page were extracted. Each extracted word has been fed to the font family classifier first to classify its font, and finally, the output determines which network to use to classify its font size (as each font has its network to recognize its size). Summary of the results with the average recognition accuracy for both font family and font size recognition can be found in Table 4.

4. CONCLUSION

In this paper, methods for font family and font size recognition have been introduced which are based on deep learning. Two systems have been introduced, 1) a font recognition system and 2) a font size recognition system. For font recognition, we used a single network to classify ten different Arabic fonts with ten different sizes (fonts supported by APTI [7]) using convolutional neural networks. The system achieved an accuracy of 98.6% on APTI dataset word images and an average recognition accuracy of 97.26% on a document generated using fonts supported by APTI and scanned with 300 Dots Per Inch (DPI). After classifying the font family of the word image, the image is fed to another convolutional neural network to classify its font size among ten different font sizes. The accuracy of font size recognition system depends on the font family; the best accuracy achieved was 99.94% for Arabic Transparent font, the lowest accuracy was 95.39% for DecoType Naskh font and the average recognition accuracy was 97.99%. The best average font size recognition accuracy for all font families on the generated and scanned document (with 300 DPI) was 98.03% for font size 8 and the lowest average recognition accuracy was 97.11% for font size 18. For all trained models, we used APTI dataset for training and testing combined with a document generated using APTI database word images and scanned with the scanner.
For the future work, the method can be generalized to support recognition of font families and font sizes of other languages.

5. REFERENCES

[1] Hamzah Luqman, Sabri A. Mahmoud and Sameh Awaida: Arabic and Farsi Font Recognition: Survey. International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI). Vol. 29, No. 1 (2015).

[2] Slimane, Fouad, Slim Kanoun, Jean Hennebert, Adel M. Alimi, and Rolf Ingold. "A study on font-family and font-size recognition applied to Arabic word images at ultra-low resolution." Pattern Recognition Letters 34, no. 2 (2013): 209-218.

[3] Jaiem, F. K., Kanoun, S., & Eglin, V. (2014, September). Arabic font recognition based on a texture analysis. In Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on (pp. 673-677). IEEE.

[4] Mousa, Mahmoud AA, Mohammed S. Sayed, and Mahmoud I. Abdalla. "An efficient algorithm for Arabic optical font recognition using scale-invariant detector." International Journal on Document Analysis and Recognition (IJDAR) 18, no. 3 (2015): 263-270.

[5] Bozkurt, Alican, Pinar Duygulu, and A. Enis Cetin. "Classifying fonts and calligraphy styles using complex wavelet transform." Signal, Image and Video Processing 9, no. 1 (2015): 225-234.

[6] Ibrahim M. Amer, Salma Hamdy, Mostafa, M. G. Mostafa, "Deep Arabic Document Layout Analysis", submitted to International conference on Intelligent Computing and Information Systems (ICICIS). Aug. 2017.

[7] APTI Arabic Printed Text Image Database. https://diuf.unifr.ch/diva/APTI/index.html

[8] Srivastava, Nitish, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. "Dropout: a simple way to prevent neural networks from overfitting." Journal of Machine Learning Research 15, no. 1 (2014): 1929-1958.

[9] Kingma, Diederik, and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[10] Nair, Vinod, and Geoffrey E. Hinton. "Rectified linear units improve restricted boltzmann machines." In Proceedings of the 27th international conference on machine learning (ICML-10), pp. 807-814. 2010.

Table 4. : Summary of the results (font family & size) of the generated document

Font Family	Font Family Recognition Accuracy	Size 6	Size 7	Size 8	Size 9	Size 10	Size 12	Size 14	Size 16	Size 18	Size 24
Advertising Bold	97.65%	98.30%	96.20%	97.39%	96.61%	95.74%	96.87%	98.01%	97.38%	99.25%	99.77%
Andalus	99.16%	97.26%	96.89%	99.58%	98.45%	99.61%	95.25%	95.96%	95.63%	95.50%	96.53%
Arabic Transparent	95.78%	95.47%	99.32%	96.78%	95.20%	99.60%	99.35%	99.50%	98.58%	96.82%	95.91%
DecoType Naskh	96.85%	99.53%	97.47%	98.78%	95.29%	98.67%	99.05%	97.41%	97.05%	98.96%	98.08%
DecoType Thuluth	98.29%	95.74%	98.68%	99.52%	99.43%	96.49%	97.58%	96.27%	98.91%	95.45%	99.05%
Diwani Letter	96.12%	97.52%	95.73%	97.87%	99.63%	95.54%	97.06%	98.46%	97.69%	95.67%	97.95%
M Unicode Sara	98.26%	95.41%	97.16%	97.11%	99.47%	99.32%	96.38%	96.71%	98.02%	98.26%	97.38%
SimplifiedArabic	97.50%	96.98%	98.58%	97.90%	95.48%	95.17%	97.86%	96.92%	97.31%	97.71%	95.46%
Tahoma	97.61%	98.41%	99.75%	98.05%	97.22%	98.84%	98.50%	95.34%	98.38%	96.27%	97.40%
TraditionalArabic	95.34%	97.29%	98.27%	97.36%	97.32%	96.82%	98.05%	99.70%	97.20%	97.23%	95.79%
Average Recognition Accuracy	97.26%	97.19%	97.80%	98.03%	97.41%	97.58%	97.60%	97.43%	97.62%	97.11%	97.33%

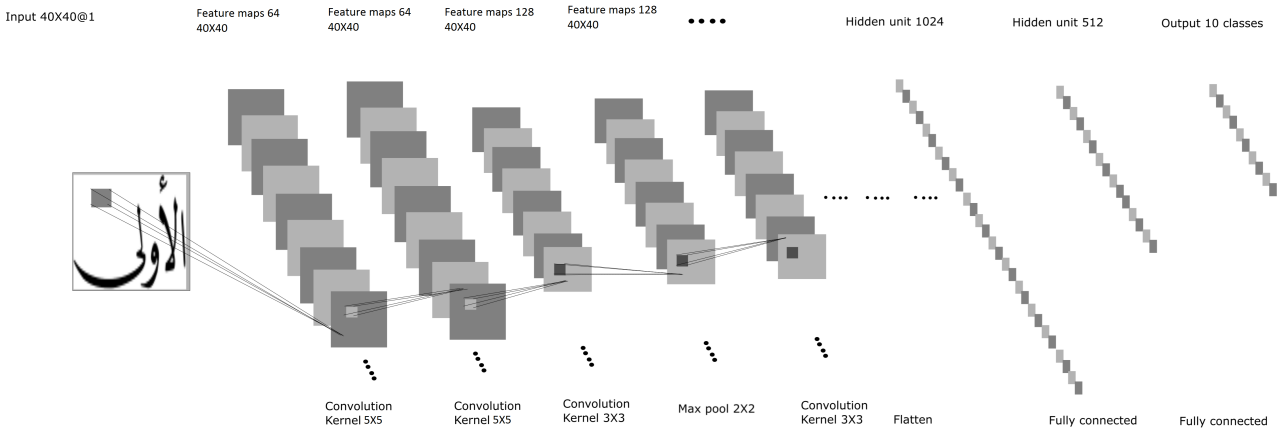

Fig. 4: Font family recognition: a demonstration of the convolutional layers and the hidden fully connected layers of the fourth network (ten font families with ten different font sizes).

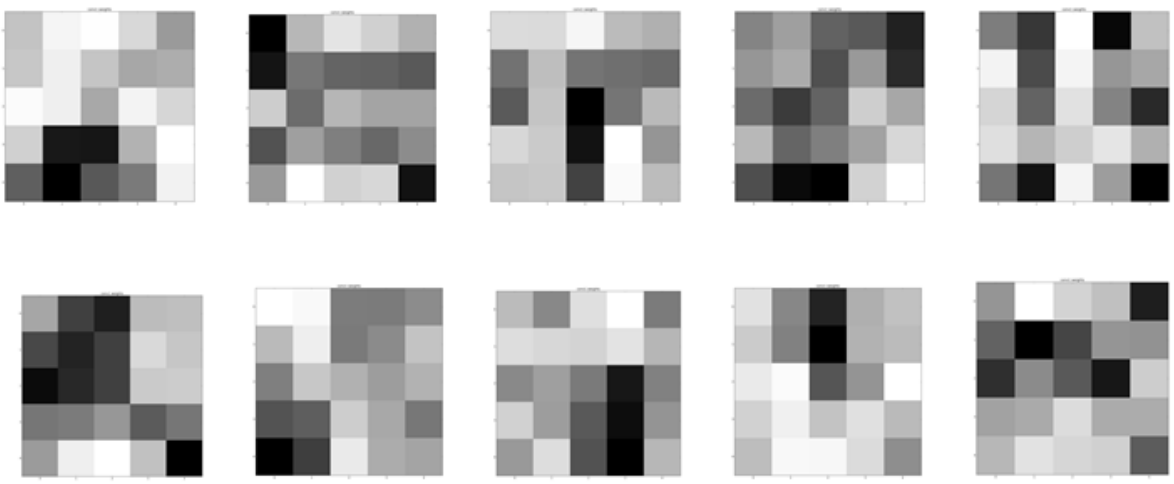

Fig. 5: Font family recognition: visualizing some convolutional filters of the first convolutional layer for the fourth network (ten font families with ten different font sizes).