Algerian Democratic and Popular Republic
Ministry of Higher Education and Scientific Research
University of Kasdi Merbah Ouargla
Faculty of New Technologies for Information and Communication

A thesis submitted in fulfilment of the requirement for the degree of

Master Academic LMD in
Fundamental Computer Science

by
Salah Belila
Youcef Gasmi

Arabic Calligraphy Recognition Using The Intrinsic Cues of Styles

---

Members of jury

Dr. Bilel Khaldi	University of Ouargla	Supervisor
Dr. Saleh Euchi	University of Ouargla	President
Dr. Oussama Aiadi	University of Ouargla	Examiner

June 2022

Dedication

We dedicate this work To our beloved parents

Abstract

This study explores the field of Arabic Calligraphy Recognition (ACR), identifies the significant limitations in the field, and devises a remedy for these limitations. Throughout our research, we found that most methods in the field of ACR rely on expert cues. Collecting and defining these cues for each calligraphy style is a tedious task, which also imposes a limit on the diversity and number of styles these approaches can cover. Therefore, we propose a new approach inspired by Eigenfaces to drop the necessity of the expert's cues, hence overcoming these limitations. The proposed approach uses Principal Component Analysis (PCA) to represent each class by a corresponding transformation matrix and then uses these matrices to predict the correct class of the input. We also collected and made publicly available a new dataset consisting of 900 images split evenly into 9 classes. This dataset is then used to validate the proposed approach. Our experiments show that the proposed approach performs better with a lower number of classes. The proposed approach scored an almost perfect accuracy in binary classification and 3-class classification, however, the accuracy gradually decreases with a higher number of classes. Our analyses linked this behaviour directly to the lack of data, which is generally a problem in the field of ACR.

Keywords: Arabic Calligraphy Automatic Recognition, Classification, Feature Extraction, PCA, Eigenfaces, Eigencalligraphies

ملخص

تقوم هذه الدراسة على استكشاف مجال التعرف على الخط العربي (ACR) ، وتحدد القيود المهمة في هذا المجال ، مع ابتكارعلاجاً لهذه القيود. خلال بحثنا ، وجدنا أن معظم الأساليب في مجال ACR تعتمد على إرشادات خبراء الخط العربي. يعد جمع هذه الإرشادات وتحديدها لكل نمط خط مهمة شاقة ، والتي تفرض أيضاً قيوداً على التنوع عدد الأنماط التي يمكن أن تغطيها هذه الأساليب. لذلك ، نقترح نهجاً جديداً مستوحى من Eigenfaces للتخلي عن ضرورة إرشادات الخبير، وبالتالي التغلب على هذه القيود. يستخدم النهج المقترح تحليل المكونات الرئيسية (PCA) لتمثيل كل فئة بواسطة مصفوفة تحويل مقابلة ثم تستخدم هذه المصفوفات للتنبؤ بالفئة الصحيحة للمدخلات. قمنا أيضاً بجمع مجموعة بيانات جديدة تتكون من 900 صورة وإتاحتها للجمهور العام ، مقسمة بالتساوي إلى 9 فئات. ثم يتم استخدام مجموعة البيانات هذه للتحقق من صحة النهج المقترح. تظهر تجاربنا أن النهج المقترح يؤدي بشكل أفضل مع عدد أقل من الفئات. حقق النهج المقترح دقة مثالية تقريباً في التصنيف الثنائي والتصنيف من فئة 3 ، ومع ذلك ، تقل الدقة تدريجياً مع وجود عدد أكبر من الفئات. ربطت تحليلاتنا هذا السلوك مباشرة بنقص البيانات ، وهي مشكلة بشكل عام في مجال ACR.

الكلمات المفتاحية : التعرف الآلي على الخط العربي، تصنيف، استخراج الميزات، تحليل المكون الرئيسي.

Contents

List of Figures

List of Tables

Chapter 1

General Introduction

In ancient times, handwriting was used as a means of communication. The poet used it to write down his poetry and poems. Scientists used it to transmit and preserve their knowledge. Nations used it to communicate with each other through messages. They also used it to convey the beliefs of their religious faith. Thanks to the importance of writing, the field of handwriting recognition has become one of the most important research topics for Pattern recognition.

Handwriting recognition (HWR) is an important issue in scientific research. Automatic recognition of text has been an active subject of research since the early days of computers. A 1972 survey cites nearly 130 works on the subject [3].

Despite being a relatively old subject, HWR remains one of the most challenging and exciting areas of research in computer science. HWR aims to simulate human reading capacity by maximizing the recognition rate. in handwriting recognition the machine interprets the user's handwritten characters or words into a format that the computer understands (e.g., Unicode text). The input device typically comprises a stylus and a touch-sensitive screen. There are many levels of HWR, starting from the recognition of simplified individual characters to the recognition of whole words and sentences of cursive handwriting [4].

The task of handwriting recognition started with relatively simple systems able to recognize single characters with very small vocabularies. The recognizers were then improved to deal with continuous handwriting in order to recognize isolated words and whole lines extracted from handwriting pages [5]. Several algorithms are used for handwriting recognition. We find machine learning algorithms such as SVM, K-Means, KNN, etc. As well as deep learning algorithms.

Given the real-life applications of handwriting recognition, it has been of great importance in the field of recognition research, among these applications:

- Bank check recognition: involves recognizing the words making up the check amount.
- Automatic mail sorting: messages are directed to the right place by recognizing the handwritten address.
- Digitization and preservation of manuscript heritage figure 1.1
- Signature verification: determining whether or not the signature is that of a given person.


بسم الله الرحمن الرحيم
وبعد حمد الله والثناء عليه والصلوة على رسوله محمد وآله فان
هذه الكتب التي صنعناها في الطب التي الاول منها هو في الاحكام
الكلية من الطب والثاني منها هو هذا الكتاب المجموع في الادوية
المفردة وقسمنا هذا الكتاب مقالتين الاول منهما في
القوانين الطبيعية التي يجب ان يعرف من امر الطب في قوى الادوية
الجزويه اما الاول فقسمناها الى ستة فصول آ في تعرف امزجة
الادويه المفردة ب في تعريف امزجة الادوية المفرده بالتجربة
ج في تعريف امزجة الادوية المفردة بالقياس د في تعريف

Figure 1.1: Historic manuscript representing the oldest copy of Avicenna's second page of "Canon of Medicine" from 1030 AD.

1.1 Calligraphy

Generally, text writing in every language can be broadly divided into two main types (Figure 1.2):

Simple fonts, which is easy to read and understand, and is mostly used in daily life among the general public. Most people tend to use this type of handwriting since anyone can understand it. For instance these types of font are used for homework, notes, diaries and to do lists etc.

The second type is calligraphy, which is more of an artistic way of handwriting. Text Calligraphy mainly involves redrawing the letters in an artistic and sometimes innovative way. Specialized tools such as the paintbrush and designated pens are often needed to perform text calligraphy. Calligraphy is often used to draw book titles, Decoration of mosques and palaces, Religious books such as the Qur'an, Art paintings. To understand it, one usually needs calligraphers experts.

Among the art of calligraphy widely spread in the world, we find Arabic calligraphy, which is the subject of our thesis.

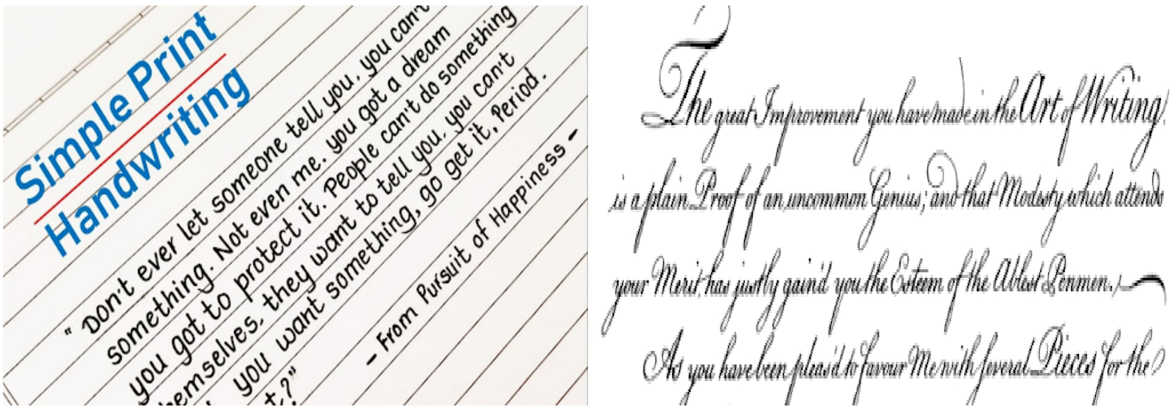


Figure 1.2: On the right, The English Calligraphy and on the left, an Simple print handwriting

1.2 Arabic Calligraphy

Arabic is the third most widely used script in the world. Around 660 million individuals use Arabic script to communicate in a number of languages, including Urdu, Pashto, Arabic, Punjabi, Persian, Malaysian, and Kurdish (to name a few) [6]. It is one of the six United Nations official languages. The emergence of Arabic calligraphy and the diversity of its forms came as a result of the flexibility of Arabic letters and their ease of flow, the clarity of their forms, and the difference between pens used for writing. The forms of Arabic calligraphy are diverse, and each calligraphy had its own writing rules.

The field of Arabic calligraphy has also expanded and branched out a lot, which made creators and those interested in this field race to create new letter forms, therefore create new calligraphy styles. This is what made the Arabic calligraphy of many styles. Among all these styles, we will study the following styles: Diwani, Naskh, Parsi, Rekaa, Thuluth, Maghribi, Kufi, Mohakek, and Square-kufic, Each style of Arabic calligraphy has characteristics and features that distinguish it from other styles. The Arabic language contains 28 letters. Each character can take more than one form according to its position in the word (at the beginning, middle, or end of the word). Table 1.1 illustrates these different forms of letter writings.

Character	Isolated	Initial	Middle	Final	Character	Isolated	Initial	Middle	Final
Alif	ا	ا	ـا	ـا	Dhad	ض	ضـ	ـضـ	ـض
Ba	ب	بـ	ـبـ	ـب	Taa	ط	طـ	ـطـ	ـط
Ta	ت	تـ	ـتـ	ـت	Dha	ظ	ظـ	ـظـ	ـظ
Tha	ث	ثـ	ـثـ	ـث	Ain	ع	عـ	ـعـ	ـع
Jeem	ج	جـ	ـجـ	ـج	Ghain	غ	غـ	ـغـ	ـغ
Ha	ح	حـ	ـحـ	ـح	Fa	ف	فـ	ـفـ	ـف
Kha	خ	خـ	ـخـ	ـخ	Qaf	ق	قـ	ـقـ	ـق
Dal	د	د	ـد	ـد	Kaf	ك	كـ	ـكـ	ـك
Thal	ذ	ذ	ـذ	ـذ	Lam	ل	لـ	ـلـ	ـل
Rai	ر	ر	ـر	ـر	Meem	م	مـ	ـمـ	ـم
Zai	ز	ز	ـز	ـز	Noon	ن	نـ	ـنـ	ـن
Seen	س	سـ	ـسـ	ـس	Ha	ه	هـ	ـهـ	ـه
Sheen	ش	شـ	ـشـ	ـش	Waw	و	و	ـو	ـو
Sad	ص	صـ	ـصـ	ـص	Ya	ي	يـ	ـيـ	ـي

Table 1.1: Characteristics of writing Arabic letters at the beginning, middle and end of a word

Arabic is written from right to left. Unlike Latin languages, the Arabic language does not contain uppercase and lowercase letters. Arabic calligraphy is also characterized by small signs called "diacritics"; These signs represent short vowels or other sounds. These marks can be either above the letter or below it. The change in diacritics changes the meaning the whole word as depicted in figure 1.3.

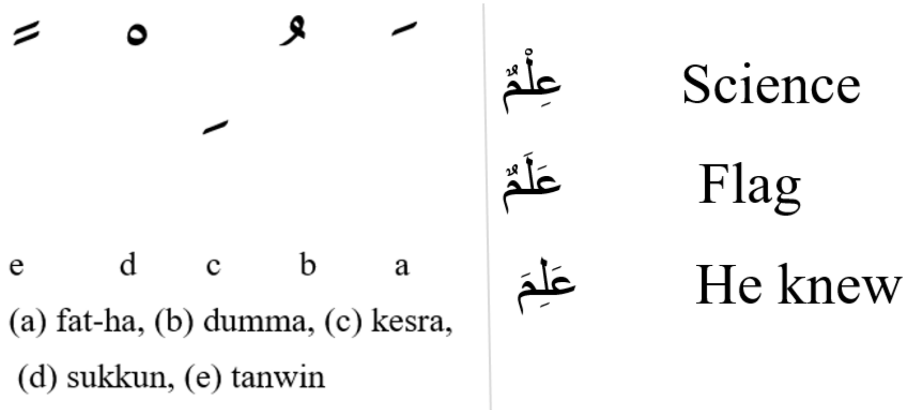


Figure 1.3: distinctive characteristics

With all these characteristics found in the Arabic language, each calligraphy style that we mentioned earlier which we will study has distinctive characteristics. Table (1.2) explains and differentiates between each of these styles.

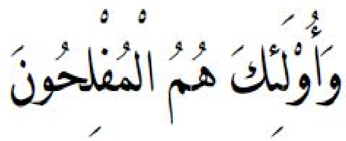
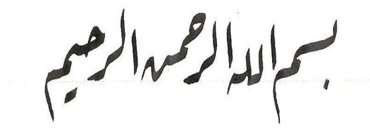
styles	Example	characteristics
• Diwani		The letters of this line are round and overlapping, it is hard to read.
• Naskh		All letters of this calligraphy have the same size. Its letters are simple. It is readable.
• Parsi		Letters are inclined from right to left in the direction from top to bottom and do not overlap and intertwine.
• Rekaa		Its letters are not overlapping. Used without diacritics. It is readable.
• Thulth		The Thuluth font is distinguished from other fonts in its structure. A single sentence can be written in several forms, depending on the structure of the letters.
• Maghribi		This font is characterized by its broad letters. It is used for writing and decoration.
• Kufi		Its letters are distinguished by their geometric angles. Some lettering is plaited.
• Mohakek		The letters of this font are decorated with points and diacritics, and the spaces between the lines are equal.
• Square-Kufic		The letters of this font are written in square and straight geometric shapes, without any curvature.

Table 1.2: Characteristics of Arabic Handwritten calligraphy

1.3 Calligraphy Style Automatic Recognition

Calligraphy style recognition is one of the applications of computer vision and pattern recognition,Automatic recognition of Arabic calligraphy is used in many applications and uses, including the following:

1. Extracting data and texts from ancient manuscripts.
2. Assist historians in classifying manuscripts and documents according to the region to which the manuscript belongs and tracing the history of these civilizations.
3. Archiving old documents and manuscripts with similar digital documents.

Calligraphy style automatic recognition is a discipline that combines image processing and machine learning techniques to process and extract information from document images, for example Define calligraphy style, Recognize letters and words.

There are many research works related to calligraphy style such as Chinese, Japanese and Arabic, and we will mention some of them below:

Chinese calligraphy There is a lot of work related to automatic recognition of Chinese calligraphy, among which we mention; Yu-Sheng Chen et al. [7] classified and predicted five Chinese handwriting styles included within a dataset of 2000 images. Each images represents one Chinese character only. The authors experimented four various machine learning algorithms looking for the algorithm that best classifies their dataset images. The experimented algorithms are Softmax regression, support vector machine, K-nearest neighbors, and random forests. Moreover, they tuned the learning hyper-parameters using the k-fold cross validation method. Jung Liang et al. [8]collected and processed the first open dataset for cursive Chinese calligraphy text, The authors presented the M6 (EM6) model as a basic model for the task of recognizing cursive Chinese calligraphy, after examining and comparing many structures of deep neural networks.

Nib calligraphy In this paper [9], SOM neural network technology is used to perform the recognition of Urdu Nastaleeq Nib Calligraphy characters, numbers from zero to nine and some symbols. Urdu Nastaleeq Nib Calligraphy are being taken from the corpus of National Language Authority Islamabad Pakistan.

The remaining parts of this thesis are structured as follows:

Chapter 1.3: In this part, We present an overview of Arabic calligraphy recognition and some related works on this topic.

Chapter 2.3: In this chapter, we will present our automated system for classifying Arabic calligraphy styles that is inspired from the Eigenfaces algorithm.

Chapter 3.4: In this chapter, we applied our automated Arabic calligraphy styles classification system to a dataset that we collected and created, We also discussed the results of experiments applying our system to the dataset.

Chapter 2

Related Work

In this section, we will present the most prominent works related to our research. We will first deal with the works related to Non-artistic Arabic calligraphy styles, and then we will address the work related to artistic AC styles, and in both parts, we will discuss the works related to two groups: the global features approach, we mean the works that are not concerned with the content and content of the image of the text, but are concerned with the style of the text in general, where we do not need to segment the image, such as classifying texts or recognizing the language. As for the second approach, it is the local features approach, by which we mean attention to the content of the text, as we need to divide the image to reach the content of the text.

2.1 Non-artistic Arabic Calligraphy Styles

There has been a lot of research on the recognition of the Arabic text and it has been widely spread, due to its importance on the one hand, and on the other hand it provides resources that help researchers to improve results and apply multiple methods. In contrast to identifying the patterns of Arabic calligraphy, which we will address next in this section. Table 2.1 provides most prominent related works, which we will discuss in two groups locale based approaches and the global based approaches.

Reference	Language	Dataset	No of font	styles
Kallel et al.(2016))[10]	Arabic	5000 block images APTID	10	2
F.slimane et al.(2013)[11]	Arabic	20000-word images APTI		
C.Tensmeyer et al.(2012))[12]	Arabic	115,068 scanned pages (KAFD)	40	
H.luqman et al.(2014) [13]	Arabic	2,576,024 Scanned Text pages	40	4
M.Lutf. et al.(2014)[14]	Arabic	privet	10	
Abuhaiba(2005)[15]	Arabic	108,000-word images	10	3
Abuhaiba(2003) [16]	Arabic	185,839 word images Privet	4	3
kallel and al(2018) [17]	Arabic	500 block images Privet	10	
Ben Moussa et al. (2010) [18]	Arabic	1000 block images	10	
Zaghden et al. (2006) [19]	Arabic	2500 block images	10	
Mousa et al. (2015) [20]	Arabic		15	
Chaker et al. (2010) [21]	Arabic	computer-generated dataset (size not indicated)	10	
Zahedi & Eslami(2011) [22]	Farisi-Arabic	1400 block images	20	
Bozkurt et al. (2015) [23]	Arabic-Farisi	60 documents	10	
Ahmed Kawther Hussein(2020) [24]	Arabic	2766 text line image	8	

Table 2.1: Arabic Font recognition related works according to [2]

In addition to the related works mentioned in Table 2.1, We will discuss one of the related works recently published (Table 2.2).

Reference	Language	Dataset	No of font	styles
Taghreed et al. (2021) [25]	Arabic	100 documents from the ALPH-REGIM database 228 documents from the KAFD database	Unknown	Unknown

Table 2.2: Arabic Font recognition related works

2.1.1 Printed Arabic Text Style Recognition Approaches Based on Global Features

Beginning with [10], who presented a solution to identify the artistic Arabic font in the global approach through two stages; image representation with a discrete curvelet transform, Back-propagation artificial neural network, The study was carried out through three different solutions; a study on high-resolution, medium-resolution and low-resolution images, on the APTID/MF dataset. The results of the study conducted on the three cases were as follows; 99.72% for high-resolution images, 99.70% for noisy images, and 99.59% for blurred images. Work [11] suggested a new method for extracting features to identify the size and type of font. The ABT dataset was used for low-resolution images, where the accuracy of font recognition reached 94.5% and the accuracy of recognition of size was 96.2%, and the accuracy of recognition of both size and font was 91.9%. [12] have used two different deep learning models (ResNet/AlexNet) to compare their performance for text font recognition of Arabic printed text (KAFD dataset) and English text (CLaMM). [15] used the decision tree to extract 48 features from 100 connected words, meaning all the letters are connected to each other, with an accuracy rate of more than 90.8%. Faten kellal used a back-propagated artificial neural network as a classifier, and used the steerable Pyramid texture descriptor for image processing.

2.1.2 Printed Arabic Text Style Recognition Approaches Based on Local Features

As for the local approach, M.lutf's [14]thesis presented a way to identify the art of calligraphy through diacritics only by the central circular projection (CCRP) to represent the image to classify calligraphy, where he depends in his work on the fragmentation of images and texts. Mousa et al. [20] proposed an algorithm that used a scale-invariant detector, k-means clustering to recognize the Arabic font in the text image, and a gradient-based descriptor. Accuracy rating between 99.2% and 99.5%. Bozkurt et al. [23] had extract features using complex wavelet transform and use support vector machines for classification. Ahmed's system relies on the Al-Ahaq method, which depends on isolated letters, by means of the difference index calculated on the polygonal approximation of the letter. For Chaker al [21],their system is based on the posterior approach that performs font recognition of isolated characters, through the difference index calculated on the polygonal approximation of the letter. In this research paper [25], a system has been proposed that uses deep learning techniques (for example RNN and LSTM) to extract knowledge from printed Arabic documents. The proposed system includes two techniques to extract keywords from document images. In the first method, extract text from the document image and then extract the word Keywords from the text. The second method is to extract the keyword from the image of the document.

2.2 Artistic Arabic Calligraphy Styles

There has been a lot of research on handwritten Arabic calligraphy. However, there was a dearth of research related to Arabic calligraphy styles, for several reasons, the most important of which are: the scarcity of resources such as data collection, the difficulty of dealing with the features of each style separately, and perhaps also one of the most important reasons, the lack of related work related to Arabic calligraphy styles. Table 2.3 presents related works Arabic calligraphy, as well as the number of patterns in the data set used. We will discuss these works in two groups, global-based approaches and local-based approaches.

Reference	Language	Dataset	No of Styles
Batainah et al.. (2012)[26]	Arabic jawi	700 block image	6
Batainah et al. (2013)[27]	Arabic jawi	700 block image	7
Batainah et al.. (2011)[28]	Arabic	4 document images	Unknown
Talab et al.(2011)[29]	Arabic	700	7
Azmi et al. (2011)[30]	Arabic jawi	100 character image	5
Allaf et al.(2016)[31]	Arabic	267 line/word Images	3
Elhmouz et al. (2020)[32]	Arabic	421 line/word Images	3

Table 2.3: Arabic calligraphy style automatic recognition related works according to [2]

In addition to the related work presented by Z Kaoudja in Table 2.3, we will also look at some related works mentioned in AC style automatic recognition, which we will present in the following table 2.4.

Reference	Language	Dataset	No. of Styles
Mohamed Ezz et al. (2019)[33]	Arabic	200 document images	2
Kalthoum Adam et al. (2017)[34]	Arabic	330 character images	6
Khayyat & Elrefaei (2020)[35]	Arabic	2653 document images	6
Z Kaoudja et al. (2021) [2]	Arabic	1685 line/word Images	9

Table 2.4: Arabic calligraphy style recognition related works

2.2.1 Arabic Calligraphy Style Recognition Approaches Based on Global features

Among the most important related works on an international level, we find Bilal Bataniah and Ahmed Talab in most of their works related to the classification of calligraphy [26, 29] A suggested statistical descriptor was used , named with Edge Direction Matrix (EDMs), 22 moments were extracted from a 3*3 matrix that counts the edges pixels' adjacency, passing through two steps EDM1 then EDM2. EDM1 for the pixels' adjacency, then EDM2 counted from EDM1, which contains the most repeated adjacency relation. Finally, the 22 moments were extracted from both EDM1 and EDM2. Another solution proposed by [32] used a common deep model (auto-encoder) for image feature extraction and a softmax layer for font type recognition. This method is able to include new fonts type without modification. Moreover, it is easy to generalize on other languages or other applications like scrip optical recognition. [33] classified and predicted only two Arabic calligraphy styles, which are Naskh and Reqaa, has been used the static Scale-Invariant Feature Transform (SIFT) and Speeded-up Robust Feature (SURF) algorithms to do the features extraction from two-hundred images. Then, they experimented with four different machine learning classifiers as follows: Gaussian naive Bayes, decision tree, random forest, and the K-nearest neighbor. They concluded that the best method for predicting the Arabic calligraphy styles is utilizing the SIFT with the Gaussian naive Bayes classifier since it recorded 92% accuracy.In [35] used a MobileNetV1 deep learning model to extract and classify the visual features in the images automatically (to classify Arabic calligraphy patterns), This model was applied to six Arabic calligraphy styles: Al-Nask, AlThulth, Al-Reqaa, Al-Hur, Al-Diwani, and Al-Farsi. The model scored 95.83% as the highest accuracy obtained.in this thesis [2], her work was based on 9 styless and a larger number of data set 1685 images, Its work is based on a multiple-classifier system for Arabic calligraphy style recognition, by extracting the basic features of each style, Although the data set is still small, it will provide an initial step for every researcher in this domain, because its data set is publicly available.

2.2.2 Arabic Calligraphy Style Recognition Approaches Based on Local Features

Researchers have been investigating local feature approaches for Arabic calligraphy styles,in [30] trying to extract the features related to the letters by dividing the letters manually. This is due to the scarcity of the data set. It is clear that this is a weak point in their work, as dividing the letters manually is a very laborious work, in addition to the fact that some letters in the Arabic calligraphy cannot be divided easily. This is because the letters are overlapping with each other. We also find [31] who presented a new method for classifying Arabic calligraphy styles, as his study relied on the form of the text, by dividing it into two halves, the bottom and the top of the baseline, focusing on the presence of diacritics and how to focus them in the text, as well as the ratio of black and white in the text. As for this method, it is not feasible for a larger number of Arabic calligraphy styles, as there is a great similarity between the Arabic calligraphy styles, in addition to the fact that this method is manual, it also represents a weak point for this work. In our study, we try to move away from manual methods and apply an automated method to classify Arabic calligraphy styles. The work [34]has focused on letters that have been segmented manually from manuscripts. Then they apply Gabor Filter (GF) to extract features. The classification is accomplished by support vector machines (SVM) and yields a recognition rate close to 82% that increased to 86.84% when local binary pattern feature vector was added to GF vector.

2.3 Conclusion

In this chapter, we discussed each of the works related to Non-artistic Arabic calligraphy styles, and artistic Arabic calligraphy styles. And by discussing the works related to automatic recognition of Arabic calligraphy, we found that the applied methods require the presence of an expert cues. This makes the number of studied Arabic calligraphy types limited.In this thesis, we will suggest a methods to dispense with the expert cues, as well as the limited number of Arabic calligraphy styles.

In the next chapter, we will present our method by which we solve the limitations of Arabic calligraphy recognition.

Chapter 3

Methodology

This chapter explains and discusses our proposed method for Arabic calligraphy style classification. We start by explaining the prerequisite concepts to understanding our approach, this mainly involves understanding Principal Component Analysis and how it is used to transform data from higher a dimension to a lower dimension and vice versa. Understanding PCA in itself is not a difficult task, however, to understand its potential and applications one must understand the intuition behind. Hence, we emphasized on explaining the intuitions behind all the introduced concepts in this chapter.

This chapter is divided into three sections. The first section explains how the PCA algorithm works, as well as, explaining the concepts needed to understand PCA. These concepts include covariance matrix, eigenvalues and eigenvectors and linear transformation using PCA. The second section is a practical example of PCA in the field of face recognition. This section discusses the eigenfaces algorithm and illustrates why it is considered an application of PCA. The eigenfaces algorithm was our main inspiration in designing our proposed approach, hence, understanding eigenfaces is directly linked to understanding our approach. Finally, the third section explains our approach by simply utilizing the knowledge acquired from the previous two sections.

3.1 Principal Component Analysis

Principal Component Analysis, or PCA, is a dimensionality-reduction method that is often used to reduce the number of dimensions of large data sets, it was invented by Karl Pearson in 1901 [36]. The result of applying PCA to a collection of points (data) in a real coordinate space is a transformation matrix. This matrix is a dictionary of p unit vectors that constitute an orthonormal basis and can be used to perform a change of basis on the data. Each vector p in the resulting matrix is called a principal component, whereas the first principal component is formed as a linear combination of the original variables (data) that explains the most variance. The second principal component explains the most variance after excluding the first principal component, and one can keep iterating until all the variance is explained. In practice, principal component analysis is commonly used to compress the data by projecting each point onto only the first few principal components. To get a general idea on how PCA works see figure 3.1

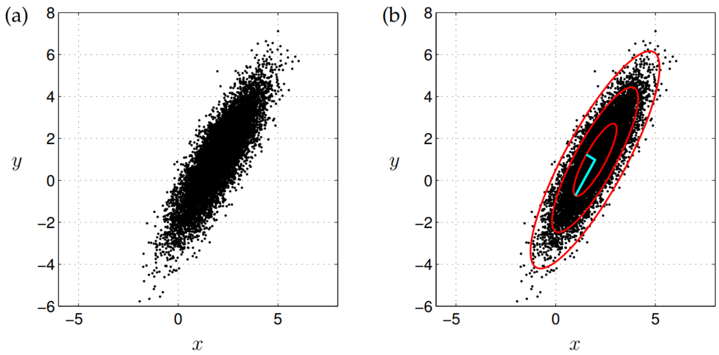


Figure 3.1: Principal components capture the variance of mean-subtracted Gaussian data. Image from [1, chapter 1]

PCA can be broken down into four steps. This section will go through each of these steps while logically explaining the purpose of each step and simultaneously providing the intuition behind PCA. These four steps are:

1. Data standardization.

2. Computing the covariance matrix.

3. Computing the eigenvalues and eigenvectors of the covariance matrix.

4. Extracting the feature vector and performing the linear transformation.

3.1.1 Data Standardization

This step is needed to standardize the real variables so that each of which contributes equally to the analysis. It is critical to perform this step because PCA is very sensitive regarding the variances of the variables. In other words, this step centers the data and brings it towards the origin.

Supposing that X is our data matrix, then each row represents a data point. We now compute the mean row (i.e. the mean of all rows) $\overline{x}$, then subtract it from X. The mean $\overline{x}$ is given by:

$$\overline{x_j} = \frac{1}{n}\sum_{i=1}^{n} X_{ij},$$

where n is the number of rows, and the mean matrix is

$$\overline{X} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \cdot \overline{x}$$

To obtain the standardized data matrix B we subtract $\overline{X}$ from X:

$$B = X - \overline{X} \tag{3.1}$$

3.1.2 The Covariance Matrix

The covariance matrix helps us understand how the data points are varying from the mean with respect to each other. This also means that the covariance matrix describes the relationships and correlations between the data points, so to identify these correlations we need to compute the covariance matrix.

The covariance matrix is a $n * n$ symmetric matrix, with n as the number of dimensions (of the data points). The covariance matrix contains the covariance of all possible pairs of the data points, the matrix below illustrates how a covariance matrix would look like.

$$\begin{bmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{bmatrix}$$

One thing to note is that the covariance of a data point with itself is equal to its variance:

$$cov(x,x) = var(x)$$

This means that the main diagonal of the covariance matrix is a vector that contains the variances of all data points. Another observation is that the covariance matrix is symmetric, and can be proven using the fact that the covariance is commutative:

$$cov(x, y) = cov(y, x)$$

Each entry in the covariance matrix (the covariance between two variables x, y) can be interpreted as the following:

- If $cov(x, y) > 0$: the two variables increase together and decrease together (correlated) the farther the value from zero is the more correlated they are.
- If $cov(x, y) < 0$: the two variables are inversely correlated (if one increases the other decreases) the farther the value from zero is the more inversely correlated they are.

One can describe the covariance matrix as a table that summarizes the correlations between every possible pair of data points.

The covariance matrix C of matrix B is computed as follows:

$$C = B^T \cdot B \tag{3.2}$$

3.1.3 Eigenvalues and Eigenvectors

Before moving on to how to compute the eigenvalues and eigenvectors, we first need to explain what they are and what they signify. Let A be a matrix, an eigenvector x of the matrix A is a vector for which the following holds:

$$Ax = \lambda x. \tag{3.3}$$

This also means that the linear transformation A on x is completely defined by λ. The value of λ decides whether the vector x is shrunk, stretched or reversed. Equation 3.3 can be rewritten as follows:

$$x(A - \lambda I) = 0 \tag{3.4}$$

where I is the identity matrix. If we assume that x in not the null-vector, then the above equation can only hold if $(A - \lambda I)$ is not invertible, meaning that the determinant of $(A - \lambda I)$ is equal to zero. Then the eigenvalues of A can computed by simply solving the equation:

$$Det(A - \lambda I) = 0 \tag{3.5}$$

The importance of eigenvectors of the covariance matrix is that they are actually the directions of the axes with the most variance (most information). In the context of principal component analysis, we call them principal components. Thus, it can be said that PCA finds directions of maximum variance of the data.

3.1.4 The Feature Vector and Linear Transformation

The feature vector is a matrix of principal components (eigenvectors) as its columns, these principal components are ordered by their corresponding eigenvalues from highest to lowest. This matrix can be used as a transformation matrix to recast the data along the principal components axis. However, in most cases, the majority of the principal components (columns of the feature vector) are discarded, and only a few principal components that explain enough variance will be preserved.

$$W = C^T \cdot B^T \tag{3.6}$$

Equation 3.6 is an example of data reduction using PCA. Where W is the projection of the standardized data matrix B into the new vector space defined by the transpose of the feature vector C^T.

3.2 Eigenfaces: A Use Case Scenario of PCA

Eigenfaces is a machine learning algorithm developed by Sirovich and Kirby [37], then first applied on the field on face classification by Matthew Turk and Alex Pentland [38]. An eigenface is the name given to an eigenvector (principal component) in the field of face recognition. Eigenfaces is the result matrix of applying PCA to a dataset of human faces. Understanding eigenfaces is crucial to understand our approach, since as mentioned earlier, our approach is heavily inspired from eigenfaces. This example serves as the stepping stone to our approach.

In this example we use the first one thousand images of the LFWcrop dataset [1] as our dataset, each image is in the size of 64x64 pixels, figure 3.2 shows a few sample images from the dataset.

[1]The dataset can be found at https://conradsanderson.id.au/lfwcrop/

Figure 3.2: Sample images from the dataset

After flattening the images (reshaping the image matrices into vectors) we obtain a dataset matrix of the shape $1000 * 4096$. Our aim is to show how to compress these images (vectors of 4096 dimensions) into much lower dimension and effectively recover these images using PCA.

Now that the dataset is ready, we can start by computing the mean row (alternatively, the mean image or the mean face), figure 3.3 shows the mean face obtained from the dataset. Then we subtract the mean face from our dataset matrix to center our data, figure 3.4 shows a few sample images of the centered dataset.

Finally, we proceed to apply the remaining steps of PCA from the previous section, the resulting matrix W is a projection of the dataset on a lower dimension, its size is $n * 1000$ where n is the number of preserved principal components. This matrix is practically a compressed version of the dataset, as well as a set of weights that will be used later with the transormation matrix from PCA to recover (approximate) the images. This stems from the fact that each face image is a linear combination of eigenfaces, we remind the reader that eigenfaces are simply principal components from the feature vector. Figure 3.5 shows the first 40 eigenfaces.

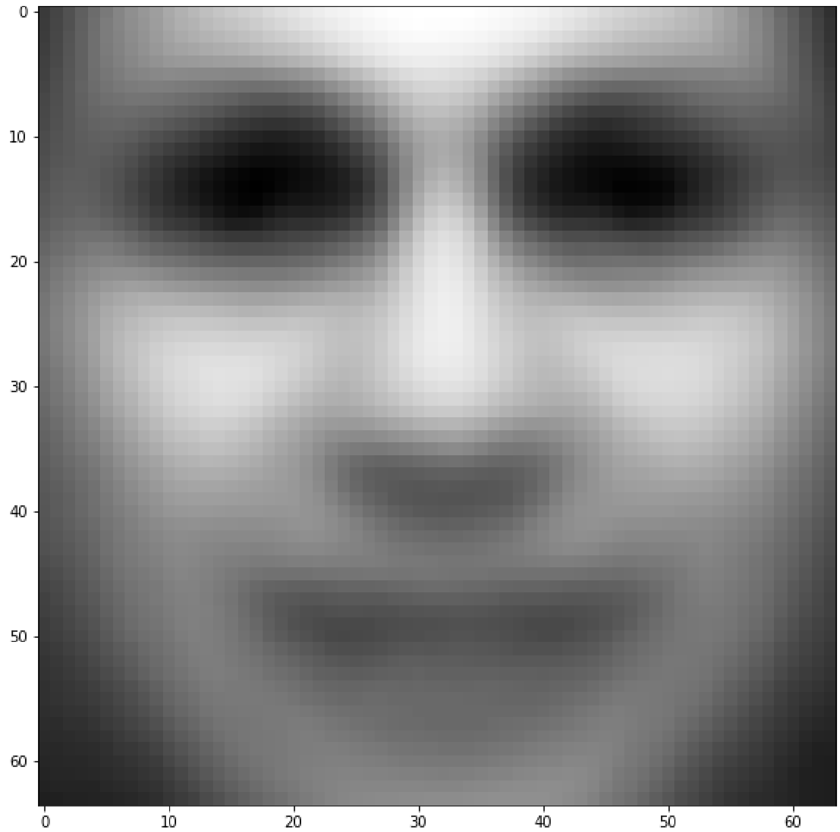


Figure 3.3: The mean face calculated from the dataset

Figure 3.4: Centered face images (after subtracting the mean face from the dataset matrix)

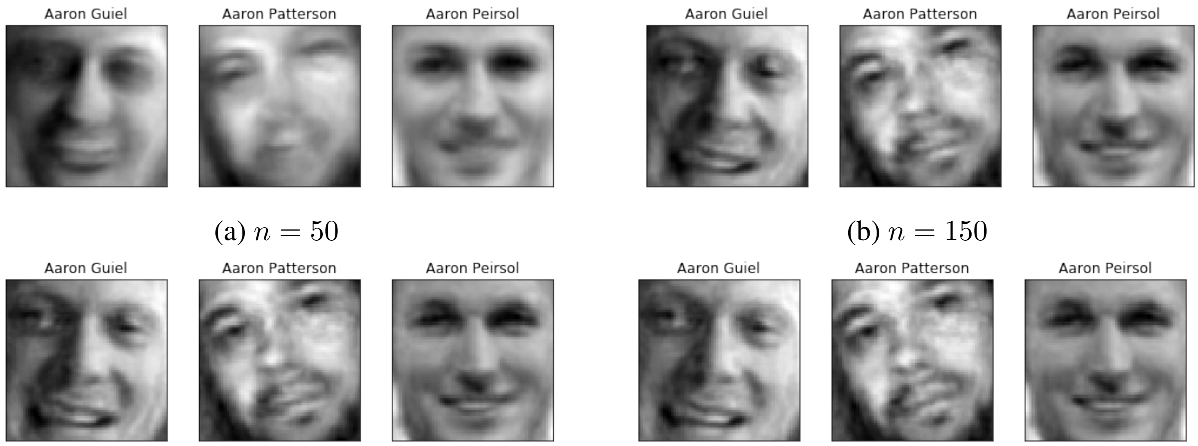
Figure 3.5: The first eigenfaces from the transformation matrix

To recover an image X, we use its corresponding weight vector W_i as follows:

$$Y = C \cdot W_i + M \tag{3.7}$$

where C is the matrix of eigenfaces, M the mean matrix and Y is the approximation of image X.

Figure 3.6 samples the results of human face approximations for different configurations of principal components n. from a simple observation, one can easily notice that higher values of n result in better approximations. However, for high enough values of n we can see that the difference is less significant (figures 3.6c and 3.6d), which is the expected behavior of PCA. it is clear that $n = 250$ yields a satisfactory result compared to the other configurations, which means that instead of storing a $4096 * 1000$ matrix, we store a $250 * 1000$ matrix (e.i. W) along with the transformation matrix (e.i. C).



(a) $n = 50$

(b) $n = 150$

(c) $n = 200$

(d) $n = 250$

Figure 3.6: The approximation of human faces for different configurations of principal components n

3.3 Our Approach: Classed Eigencalligraphies

Throughout our research we found that the major limitations of the current Arabic calligraphy style recognition systems are:

1. They are heavily dependent on the expert(s) cues.

2. The number of classes that they can recognize is limited.

Our approach is a classification algorithm that predicts the calligraphy style of an input image, this approach is designed to overcome the aforementioned limitations.

We split our dataset into several matrices where each matrix contains images from only one class, then we applied PCA to each of these matrices to obtain a transformation matrix for each class. This means each class can be represented by its corresponding transformation matrix. The transformation matrix of a class holds the most significant features of the corresponding calligraphy style. We denote the transformation matrix of the class c_i as C_i.

Next, for an input X, we transform X similar to the example of eigenfaces from the previous section. However, in this case we do this for each matrix C_i, which will produce n projections for image X where n is the number of classes. we denote P_i as the projection of X using C_i.

Finally, we try to approximate X using P_i and C_i similar to the example of eigenfaces from the previous section. The result is a set of candidate approximations of X, we denote them as Y_i.

The transformation matrix of the correct class should produce the best approximation since it was trained on similar data. Hence, we compute the distance between X and each candidate Y_i and pick the closest candidate approximation Y_k as our best candidate and therefore, c_k as the predicted class. Figure 3.7 shows a summarized depiction of our approach.

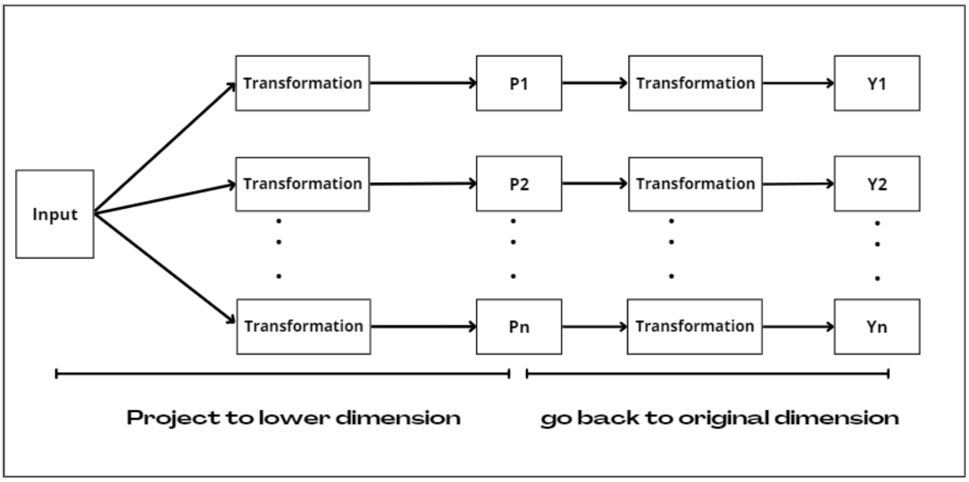


Figure 3.7: An illustration of our approach

This approach uses the features extracted by PCA as its cues, which drops the necessity of the expert's cues. It also does not impose a limit on the number of classes. we named this approach Classed Eigencalligraphies. Next is the formal definition of our approach.

Let X a column vector be the input, and C_i the transformation matrix representing the class c_i, then the projection of X, P_i is defined as follows:

$$P_i = C_i^T \cdot X \tag{3.8}$$

then Y_i the candidate approximation of X for class c_i is computed as follows:

$$Y_i = C_i \cdot P_i + M \tag{3.9}$$

where M is the mean matrix. Then we can obtain the index p of the predicted class c_p by the following equation:

$$p = \arg\min \begin{bmatrix} f(X, Y_1) \\ \vdots \\ f(X, Y_n) \end{bmatrix} \tag{3.10}$$

where n is the number of classes and f is the distance function[2].

[2]Euclidean distance was used in the experiments

3.4 Conclusion

This chapter explained the approach we are going to use in tackling the problem of Arabic calligraphy style recognition, and also explained the mathematics behind it. In the next chapter we will see the effectiveness of our approach by applying it on a dataset that we collected and created.

Chapter 4

Experiments and Results

In this chapter we will explain the conducted experiments, our dataset and setup, then discuss the results of these experiments. Our experiments show that our proposed approach is viable, and also shows that it did overcome the limitations of other approaches hinted throughout this thesis (the necessity of the expert and the limit on the number of classes). We also exhibit our new dataset in this chapter, then proceed to discuss the method we used to collect this data.

This chapter is divided into two main sections as follows:

1. **Data collection:** this section discusses the methods we used to collected our dataset and shows a few samples of the dataset.

2. **Experiments and results:** this section discusses how the experiments were conducted and the setup of these experiment, then shows the results and observation and remarks on these results.

4.1 Data Collection

An issue that researchers in the field of Arabic calligraphy classification the unavailability of data, mainly due to the lack of related work. Therefore, we collected and made publicly available[1] a new dataset for Arabic calligraphy classification.

The dataset constitutes of 900 grayscale images of 100x100 pixels, these images were obtained after cropping the images of the dataset proposed in [39]. The dataset contains 100 images for each of the nine calligraphy styles (Diwani, Naskh, Parsi, Rekaa, Thulth, Maghribi, Kufi, Mohakek and Square-Kufic). Figure 4.1 shows sample images from the dataset for each calligraphy style.

(a) Diwani

(b) Naskh

(c) Parsi

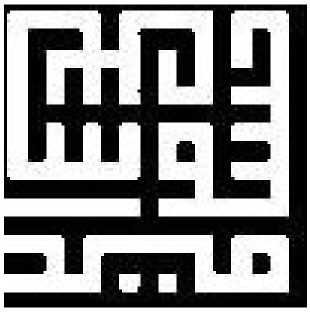
(d) Rekaa

(e) Thuluth

(f) Maghribi

(g) Kufi

(h) Mohakek



(i) Square-Kufic

Figure 4.1: Image samples of each calligraphy style in the dataset

The cropping of the dataset images was made in a way that allows the image to contain as

[1]The dataset can be found at: https://drive.google.com/file/d/1WNFYiAHcCemTZtdsjbTWVh9$_{WgZSzlbx/}$

much features of the calligraphy style as possible, as well as spatial information which allows to capture the correlations between the image segments (diacritics, letter shape, style-specific pen strokes, etc.).

4.2 Experiments and Results

We trained our algorithm using the dataset we proposed in the previous section. The training set constituted of 80 images for each class, against 20 test images for each class (both image sets are converted to gray level). In this section we will observe and discuss the results of our algorithm starting with a low number of classes and increasing the number until all the classes of the dataset are covered.

We start with 2-classes as shown in figure 4.2, we see that our algorithm scored relatively high in a binary classification. This proves that our approach is effectively applicable on the Arabic calligraphy style classification problem.

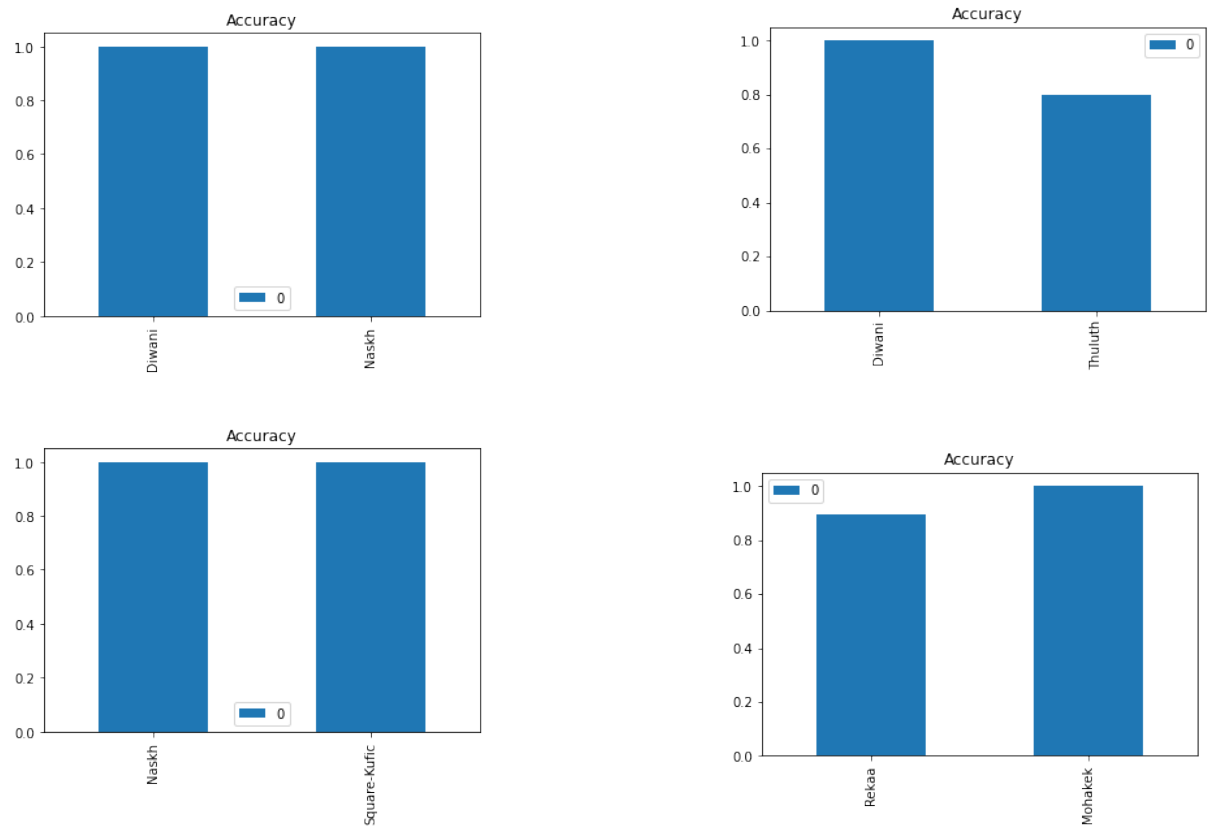


Figure 4.2: The accuracy of using classed eigencalligraphies for binary calligraphy style classification

Figure 4.3 shows the results for three, five and six classes, we can see that the accuracy for Kufic and Maghribi calligraphies is significantly lower compared to the accuracy for other calligraphy style classes. Furthermore, their accuracy keeps decreasing as the number of classes increases. The cause of this behavior is the lack of training data, and the next paragraph proves this statement.

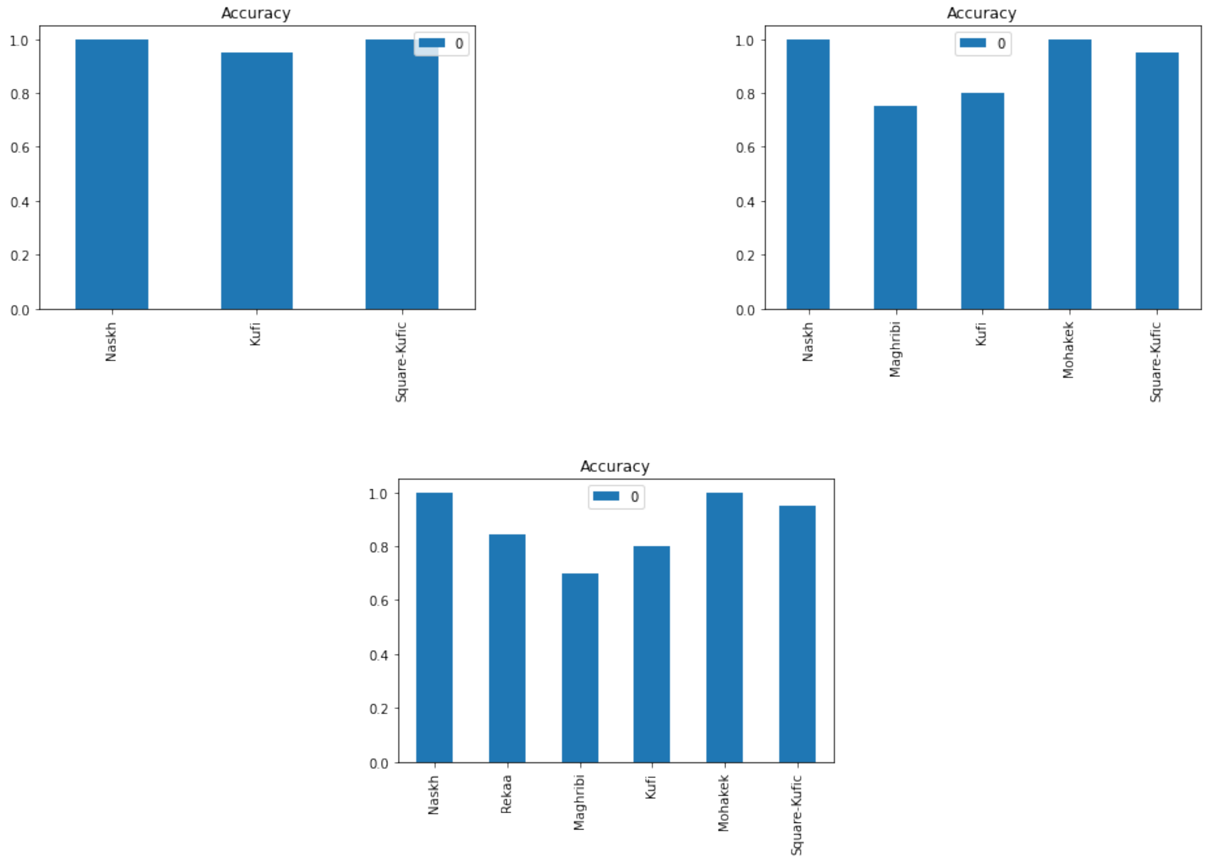


Figure 4.3: The accuracy of using classed eigencalligraphies for multi-class calligraphy style classification

In our experiments we noticed that in the case of binary classification, the algorithm did not need the whole 80 images, but only 53 images were sufficient to produce the best results (we know it is the best for our algorithm because the results did not change even when we provided more than 53 images). Similarly, only 68 training images were needed for the 3-class classification. However, in the cases of more classes the algorithm could not reach its maximum potential (this is because we kept increasing the number of training images and the accuracy kept increasing until we hit the 80 images limit).

Figure 4.4 shows the results of applying classed eigencalligraphies on the whole dataset and we also found that fixing the number of principal components for all classes to $n = 20$ results in the best general accuracy equal to 60%.

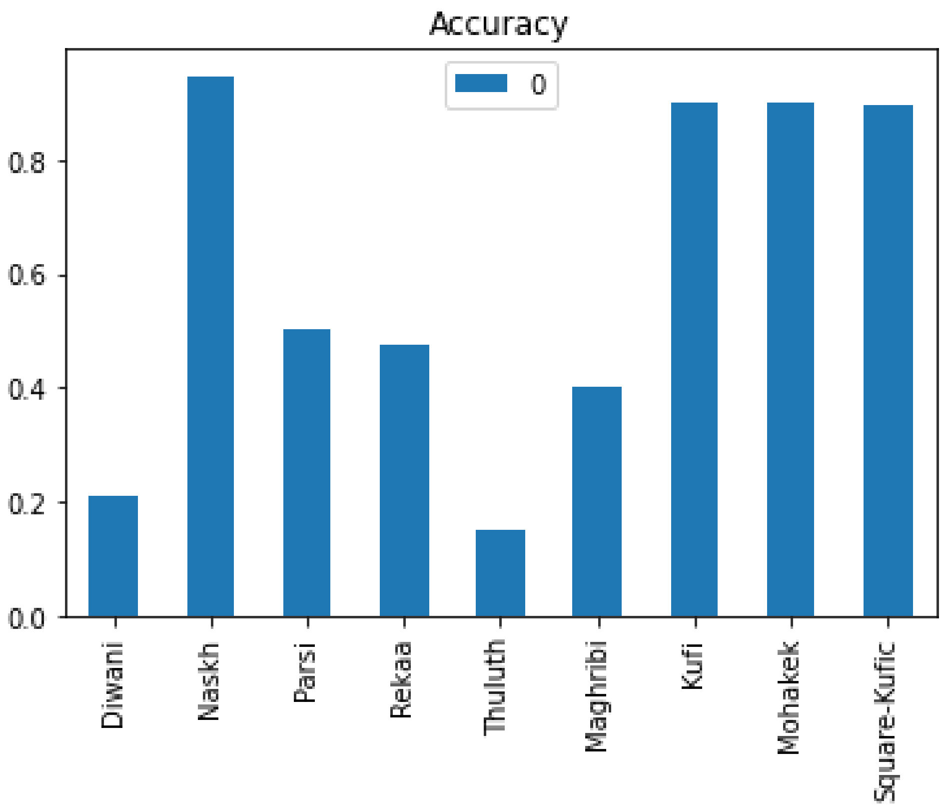


Figure 4.4: The results of applying classed eigencalligraphies on the whole dataset

Calligraphy class	Our Method	State of the art [2]
Diwani	21%	98%
Naskh	95%	99%
Parsi	50%	97%
Rekaa	48%	98%
Thuluth	15%	94%
Maghribi	40%	97%
Kufi	90%	98%
Mohakek	90%	94%
Square-Kufic	89%	99%

Table 4.1: A comparison between our method and the state of the art results

Table 4.1 compares the results yielded by our method and the results of the state of the art. We can see that both methods scored their lowest in Thuluth. And Square-Kufic, Kufi and Naskh being the most distinguishable among all the calligraphy styles. However, the state of the art also scored its lowest in the Mohakek, whereas our method scored its second-highest score.

	Diwani	Naskh	Parsi	Rekaa	Thuluth	Maghribi	Kufi	Mohakek	S-Kufic
Diwani	21%	0%	42%	16%	0%	0%	0%	21%	0%
Naskh	0%	95%	0%	0%	0%	5%	0%	0%	0%
Parsi	5%	15%	50%	30%	0%	0%	0%	0%	0%
Rekaa	5%	5%	37%	47%	0%	5%	0%	0%	0%
Thuluth	10%	5%	20%	20%	15%	0%	10%	0%	20%
Maghribi	5%	5%	0%	0%	0%	40%	50%	0%	0%
Kufi	0%	0%	0%	0%	0%	5%	90%	0%	5%
Mohakek	0%	10%	0%	0%	0%	0%	0%	90%	0%
S-Kufic	0%	0%	11%	0%	0%	0%	0%	0%	89%

Table 4.2: The confusion matrix of the nine studied styles generated by our approach

Table 4.2 illustrates the confusion matrix generated by our approach. We can see that our approach produces satisfying results with less artistic calligraphy styles, namely, Naskh, Kufi, Mohakek, and Square-Kufic. We also notice that our method confuses visually similar styles with each other, Diwani, Parsi, Rekaa and Naskh and Mohakek. While Thuluth is confused with most other calligraphy styles because it has traits from most of the other styles, especially the curves (similar to Parsi, Rekaa and Diwani). This is the expected behaviour because our approach captures the most common features of the calligraphy styles (using PCA) and bases its prediction upon these features.

4.3 Conclusion

In this chapter we proposed a new dataset for calligraphy style classification, while pointing out the lack of datasets in this field of research. We also used this dataset in our experiments with our new approach: Classed Eigencalligraphies, and proved that the proposed dataset is fit for Arabic calligraphy style classification. Our experiments and results also showed that Classed Eigencalligraphies algorithm can be used to effectively tackle the problem of Arabic calligraphy style classification given enough training data.

Conclusion

In this study we researched several works in the field of Arabic calligraphy recognition, and found various limitations in the current approaches used in the field of Arabic calligraphy style classification. We concluded that the major limitations we found are:

1. The necessity of the expert's cues in several approaches.

2. The limit on the number of classes in the majority of approaches.

To address these problems, we proposed a new approach inspired from Eigenfaces, namely, Classed Eigencalligraphies and we proved through our experiments that classed eigenfaces was able to overcome the limitation mentioned above.

We also created and made publicly available a new dataset for Arabic calligraphy style recognition, this dataset was then used in our experiments.

We noticed from our study that the field of Arabic calligraphy recognition still does not have standard and established datasets, and the existing data is much less compared to other fields of computer vision and artificial intelligence in general, and the datasets in this field of research are relatively smaller compared to the sizes of datasets in other fields of artificial intelligence. Hence, future work can start by augmenting existing datasets in the field of Arabic calligraphy recognition, or creating new ones.

Bibliography

[1] J. Nathan Kutz Steven L. Brunton. *Data Driven Science Engineering Machine Learning, Dynamical Systems, and Control*. 2017.

[2] Zineb Kaoudja, Mohammed Lamine Kherfi, and Belal Khaldi. A new computational method for arabic calligraphy style representation and classification. *Applied Sciences*, 11:4852, 05 2021.

[3] L.D. Harmon. Automatic recognition of print and script. *Proceedings of the IEEE*, 60(10):1165–1176, 1972.

[4] Miika Silfverberg. Chapter 1 - historical overview of consumer text entry technologies. In I. Scott MacKenzie and Kumiko Tanaka-Ishii, editors, *Text Entry Systems*, pages 3–25. Morgan Kaufmann, Burlington, 2007.

[5] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez, Sergi Mestre, Juan mas romeu, David Mota, Jon Almazan, and Lluís-Pere Heras. Icdar 2013 robust reading competition. pages 1484–1493, 08 2013.

[6] Pariona Amber. The world's most popular writing scripts.

[7] Yu-Sheng Chen, Su, and Li. Machine learning for calligraphy styles recognition.

[8] Jung Liang, Wen-Hung Liao, and Yi-Chieh Wu. Toward automatic recognition of cursive chinese calligraphy : An open dataset for cursive chinese calligraphy text. In *2020 14th*

International Conference on Ubiquitous Information Management and Communication (IMCOM). IEEE, January 2020.

[9] Mahdi Hamdani, Amr El Desoky Mousa, and Hermann Ney. Open vocabulary arabic handwriting recognition using morphological decomposition. pages 280–284, 2013.

[10] Faten Kallel Jaiem and Monji Kherallah. A novel arabic font recognition system based on texture feature and dynamic training. *Int. j. intell. syst. technol. appl.*, 16(4):289, 2017.

[11] Fouad Slimane, 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 Recognit. Lett.*, 34(2):209–218, January 2013.

[12] Chris Tensmeyer, Daniel Saunders, and Tony Martinez. Convolutional neural networks for font classification. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, November 2017.

[13] Hamzah Luqman, Sabri A Mahmoud, and Sameh Awaida. KAFD arabic font database. *Pattern Recognit.*, 47(6):2231–2240, June 2014.

[14] Mohammed Lutf, Xinge You, Yiu-Ming Cheung, and C L Philip Chen. Arabic font recognition based on diacritics features. *Pattern Recognit.*, 47(2):672–684, February 2014.

[15] I Abuhaiba. Arabic font recognition based on templates. *The International Arab*, 2003.

[16] Ibrahim S I Abuhaiba. Arabic font recognition using decision trees built from common words. *J. Comput. Inf. Technol.*, 13(3):211, 2005.

[17] Faten Kallel, Anis Mezghani, Slim Kanoun, and Monji Kherallah. Arabic font recognition based on discret curvelet transform. In *Proceedings of the Third International Afro-European Conference for Industrial Advancement — AECIA 2016*, Advances in intelligent systems and computing, pages 360–369. Springer International Publishing, Cham, 2018.

[18] Sami Ben Moussa, Abderrazak Zahour, Abdellatif Benabdelhafid, and Adel M Alimi. New features using fractal multi-dimensions for generalized arabic font recognition. *Pattern Recognit. Lett.*, 31(5):361–371, April 2010.

[19] N Zaghden, S B Moussa, and A M Alimi. Reconnaissance des fontes arabes par. 2006.

[20] Mahmoud A A Mousa, Mohammed S Sayed, and Mahmoud I Abdalla. An efficient algorithm for arabic optical font recognition using scale-invariant detector. *Int. J. Doc. Anal. Recognit.*, 18(3):263–270, September 2015.

[21] I Chaker, M Harti, H Qjidah, and R Benslimane. *Recognition of Arabic characters*. 2010.

[22] Morteza Zahedi and Saeideh Eslami. Farsi/Arabic optical font recognition using SIFT features. *Procedia Comput. Sci.*, 3:1055–1059, 2011.

[23] Alican Bozkurt, Pinar Duygulu, and A Enis Cetin. Classifying fonts and calligraphy styles using complex wavelet transform. *Signal Image Video Process.*, 9(S1):225–234, December 2015.

[24] Ahmed Kawther Hussein. Fast learning neural network based on texture for arabic calligraphy identification. *Indones. j. electr. eng. comput. sci.*, 21(3):1794, March 2021.

[25] Taghreed Alghamdi, Samia Snoussi, and Lobna Hsairi. Deep learning arabic printed document knowledge extraction. In *The 5th International Conference on Future Networks & Distributed Systems*, pages 201–207, 2021.

[26] Bilal Bataineh, Siti Norul Huda Sheikh Abdullah, and Khairuddin Omar. A novel statistical feature extraction method for textual images: Optical font recognition. *Expert Syst. Appl.*, 39(5):5470–5477, April 2012.

[27] Bilal Bataineh, Siti Norul Huda Sheikh Abdullah, Khairuddin Omar, and Anas Batayneh. Arabic-jawi scripts font recognition using first-order edge direction matrix. In *Soft Computing Applications and Intelligent Systems*, Communications in computer and information science, pages 27–38. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[28] Bilal Bataineh, Siti Norul Huda Sheikh Abdullah, and Khairudin Omar. Arabic calligraphy recognition based on binarization methods and degraded images. In *2011 International Conference on Pattern Analysis and Intelligence Robotics*. IEEE, June 2011.

[29] Mohammed Ahmed Talab, Siti Norul Huda Sheikh Abdullah, and Mohammad Hakim Assiddiq Razalan. Edge direction matrixes-based local binary patterns descriptor for invariant

pattern recognition. In *2013 International Conference on Soft Computing and Pattern Recognition (SoCPaR)*. IEEE, December 2013.

[30] Mohd Sanusi Azmi, Khairuddin Omar, Mohammad Faidzul Nasrudin, Khadijah Wan Mohd Ghazali, and Azizi Abdullah. Arabic calligraphy identification for digital jawi paleography using triangle blocks. In *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics*. IEEE, July 2011.

[31] S Allaf and R Al-Hmouz. Automatic recognition of artistic arabic calligraphy types. - , 27(1), 2016.

[32] R Al-Hmouz. *Deep learning autoencoder approach: Automatic recognition of artistic*.

[33] Mohamed Ezz, Mohamed, and Al-Amira. Classification of arabic writing styles in ancient arabic manuscripts. *Int. J. Adv. Comput. Sci. Appl.*, 10(10), 2019.

[34] Kalthoum Adam, Somaya Al-Maadeed, and Ahmed Bouridane. Letter-based classification of arabic scripts style in ancient arabic manuscripts: Preliminary results. In *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*. IEEE, April 2017.

[35] Manal Khayyat and Lamiaa Elrefaei. A deep learning based prediction of arabic manuscripts handwriting style. *International Arab Journal of Information Technology*, 17:702–712, 9 2020.

[36] Karl Pearson. LIII. ion lines and planes of closest fit to systems of points in space/i. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, November 1901.

[37] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 4(3):519, March 1987.

[38] M.A. Turk and A.P. Pentland. Face recognition using eigenfaces. In *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Comput. Sco. Press.

[39] Zineb Kaoudja, Mohammed Lamine Kherfi, and Belal Khaldi. An efficient multiple-classifier system for arabic calligraphy style recognition. In *2019 International Conference on Networking and Advanced Systems (ICNAS)*. IEEE, June 2019.