A FRAMEWORK FOR ANCIENT AND MACHINE-PRINTED MANUSCRIPTS CATEGORIZATION

by

Ehsan ARABNEJAD

MANUSCRIPT-BASED THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE IN PARTIAL FULFILLMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
Ph.D.

MONTREAL, JANUARY 24, 2018

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Mohamed Cheriet, Thesis Supervisor
Department of génie de la production automatisée, École de technologie supérieure

Ms. Sylvie Ratté, President of the Board of Examiners
Department of génie logiciel et des technologies de l'information, École de technologie supérieure

Mr. Luc Duong, Member of the jury
Department of génie logiciel et des technologies de l'information, École de technologie supérieure

Mr. Ching Yee Suen, External Independent Examiner
Department of computer science and software engineering , Concordia university

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON JANUARY 16, 2018

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

I would like to thank my advisor, Mohamed Cheriet for guiding and supporting me over the years. You have set an example of excellence as a researcher, mentor and instructor. I would like to thank my thesis committee members for all of their guidance through this process; your discussion and feedback have been absolutely invaluable. I would especially like to thank my amazing family for the love, support, and constant encouragement I have gotten over the years. I would also like to thank my friends and my colleagues from Synchromedia Lab.

This research was realized thanks to financial support of the Natural Science and Engineering Research Council of Canada (NSERC).

UN CADRE POUR LA CATÉGORISATION DES MANUSCRITS ANCIENS ET IMPRIMÉS

Ehsan ARABNEJAD

RÉSUMÉ

La compréhension de l'image documentaire (DIU) a attiré beaucoup d'attention et est devenue l'un des domaines de recherche actifs. Bien que le but ultime de DIU soit d'extraire des informations textuelles d'une image de document, de nombreuses étapes sont impliquées dans un tel processus tel que la catégorisation, la segmentation et l'analyse de mise en page. Toutes ces étapes sont nécessaires pour obtenir un résultat précis à partir de la reconnaissance de caractères ou de la reconnaissance de mots d'une image de document. L'une des étapes importantes dans DIU est la catégorisation d'image de document (DIC) qui est nécessaire dans de nombreuses situations telles que l'image de document multi-script ou multi-polices ou multi-langue. Cette étape fournit des informations utiles pour le système de reconnaissance et aide à réduire son erreur en permettant d'incorporer un système de reconnaissance optique de caractères (OCR) ou un système de reconnaissance de mots (WR) spécifique à une catégorie. Cette recherche se concentre sur le problème de DIC dans différents niveaux de script, de style et de langage et établit un cadre pour la représentation flexible et l'extraction de caractéristiques qui peuvent être adaptées à de nombreux problèmes DIC. Les méthodes actuelles pour DIC ont de nombreuses limitations et inconvénients qui limitent l'utilisation pratique de ces méthodes. Nous proposons des nouvelles méthodes de catégorisation de l'image de document en fonction de la représentation des patches et la «Non-negative Matrix Factorization (NMF)».

De nombreuses méthodes existent pour l'identification par script de l'image du document, mais peu d'entre elles ont abordé le problème dans les manuscrits imprimés et elles ont beaucoup de limites et d'inconvénients. Par conséquent, notre premier objectif est d'introduire une nouvelle méthode pour l'identification des manuscrits anciens. La méthode proposée est basée sur une représentation de patches dans laquelle les patches sont extraits à l'aide de la carte squelette d'une image de document. Cette représentation surmonte la limitation des méthodes actuelles sur le niveau spécifique de mise en page. Le schéma proposé d'extraction de caractéristiques basé sur la «Projective Non-negative Matrix Factorization (PNMF)» est robuste contre les variations de bruit et d'écriture et peut être utilisé pour différents scripts. La méthode proposée est plus performante que les méthodes de pointe et peut être appliquée à différents niveaux de mise en page.

Les méthodes actuelles d'identification de police de caractères (ou style d'écriture) sont principalement proposées pour être appliquées sur une image de document imprimée à la machine et beaucoup d'entre elles ne peuvent être utilisées que pour une niveau de mise en page spécifique. Par conséquent, nous proposons une nouvelle méthode pour l'identification de police et de style des manuscrits imprimés et manuscrits basés sur la représentation de patch et «Non-negative Matrix Tri-Factorization (NMTF)». Les images sont représentées par des patches superposés qui ont obtenus à partir des pixels de premier plan. La position de ces patches est définie en fonction de la carte du squelette afin de réduire le nombre de patches. NMTF est utilisée pour apprendre les bases de chaque police de caractères (ou style), puis ces bases sont utilisées pour classer une nouvelle image basée sur de erreur de représentation minimale. La méthode proposée peut facilement être étendue à des nouvelles polices de caractères car les bases de chaque police de caractères sont apprises séparément des autres polices de caractères. Cette méthode est testée sur deux bases de données de manuscrits anciens et imprimés à la machine et les résultats ont confirmé sa performance par rapport aux méthodes de pointe.

Enfin, nous proposons une nouvelle méthode pour l'identification de la langue de document basée sur la représentation de patch et Non-negative Matrix Tri-Factorization (NMTF). Les méthodes d'identification du langage sont basées sur des données textuelles obtenues par le moteur OCR ou des données d'images par codage et comparaison avec des données textuelles. La méthode basée sur le moteur OCR nécessite beaucoup de traitement et la méthode basée sur l'image actuelle ne s'applique pas aux scripts cursifs tels que l'Arabe. La représentation du patch fournit le composant du script Arabe (lettres) qui ne peut pas être extrait simplement par des méthodes de segmentation. NMTF est utilisé pour l'apprentissage du dictionnaire et la génération de «codebook» qui seront utilisés pour représenter l'image du document avec un histogramme. La méthode proposée est testée sur deux séries de manuscrits ancients et imprimés et comparée aux caractéristiques n-gram obtenues (basées sur le texte) et aux caractéristiques de textures et de codebook (basées sur l'image) pour valider la performance.

Les méthodes proposées sont robustes a la variation des écritures, les changements dans le polices de caractères (ou style d'écriture) et la présence de dégradation. Elles sont aussi flexibles quant aux différents niveaux de mise en page (d'une ligne de texte à un paragraphe). Les méthodes de cette recherche ont été testées sur des ensembles de manuscrits et imprimés à la machine et comparées à des autres méthodes. Toutes les évaluations montrent l'efficacité, la robustesse et la flexibilité des méthodes proposées pour la catégorisation de l'image de document. Comme mentionné précédemment, les stratégies proposées fournissent un cadre pour une représentation efficace et flexible et une extraction de caractéristiques pour la catégorisation d'images de documents. Ce cadre de travail peut être appliqué à différents niveaux de mise en page, les informations provenant de différents niveaux de mise en page peuvent être fusionnées et mélangées et ce cadre peut être étendu à des situations plus complexes et à des problems différentes.

Mots-clés: Catégorisation d'image de document, Clustering, Non-Negative Matrix Factorization, Identification de script, Identification de police de caracterés, Identification de langage de document

A FRAMEWORK FOR ANCIENT AND MACHINE-PRINTED MANUSCRIPTS CATEGORIZATION

Ehsan ARABNEJAD

ABSTRACT

Document image understanding (DIU) has attracted a lot of attention and became an of active fields of research. Although, the ultimate goal of DIU is extracting textual information of a document image, many steps are involved in a such a process such as categorization, segmentation and layout analysis. All of these steps are needed in order to obtain an accurate result from character recognition or word recognition of a document image. One of the important steps in DIU is document image categorization (DIC) that is needed in many situations such as document image written or printed in more than one script, font or language. This step provides useful information for recognition system and helps in reducing its error by allowing to incorporate a category-specific Optical Character Recognition (OCR) system or word recognition (WR) system. This research focuses on the problem of DIC in different categories of scripts, styles and languages and establishes a framework for flexible representation and feature extraction that can be adapted to many DIC problem. The current methods for DIC have many limitations and drawbacks that restrict the practical usage of these methods. We proposed an efficient framework for categorization of document image based on patch representation and Non-negative Matrix Factorization (NMF). This framework is flexible and can be adapted to different categorization problem.

Many methods exist for script identification of document image but few of them addressed the problem in handwritten manuscripts and they have many limitations and drawbacks. Therefore, our first goal is to introduce a novel method for script identification of ancient manuscripts. The proposed method is based on patch representation in which the patches are extracted using skeleton map of a document images. This representation overcomes the limitation of the current methods about the fixed level of layout. The proposed feature extraction scheme based on Projective Non-negative Matrix Factorization (PNMF) is robust against noise and handwriting variation and can be used for different scripts. The proposed method has higher performance compared to state of the art methods and can be applied to different levels of layout.

The current methods for font (style) identification are mostly proposed to be applied on machine-printed document image and many of them can only be used for a specific level of layout. Therefore, we proposed new method for font and style identification of printed and handwritten manuscripts based on patch representation and Non-negative Matrix Tri-Factorization (NMTF). The images are represented by overlapping patches obtained from the foreground pixels. The position of these patches are set based on skeleton map to reduce the number of patches. Non-Negative Matrix Tri-Factorization is used to learn bases from each fonts (style) and then these bases are used to classify a new image based on minimum representation error. The proposed method can easily be extended to new fonts as the bases for each font are learned separately from the other fonts. This method is tested on two datasets of machine-printed and ancient manuscript and the results confirmed its performance compared to the state of the art methods.

Finally, we proposed a novel method for language identification of printed and handwritten manuscripts based on patch representation and Non-negative Matrix Tri-Factorization (NMTF). The current methods for language identification are based on textual data obtained by OCR engine or images data through coding and comparing with textual data. The OCR based method needs lots of processing and the current image based method are not applicable to cursive scripts such as Arabic. In this work we introduced a new method for language identification of machine-printed and handwritten manuscripts based on patch representation and NMTF. The patch representation provides the component of the Arabic script (letters) that can not be extracted simply by segmentation methods. Then NMTF is used for dictionary learning and generating codebooks that will be used to represent document image with a histogram. The proposed method is tested on two datasets of machine-printed and handwritten manuscripts and compared to n-gram features (text-based), texture features and codebook features (image-based) to validate the performance.

The above proposed methods are robust against variation in handwritings, changes in the font (handwriting style) and presence of degradation and are flexible that can be used to various levels of layout (from a textline to paragraph). The methods in this research have been tested on datasets of handwritten and machine-printed manuscripts and compared to state-of-the-art methods. All of the evaluations show the efficiency, robustness and flexibility of the proposed methods for categorization of document image. As mentioned before the proposed strategies provide a framework for efficient and flexible representation and feature extraction for document image categorization. This frame work can be applied to different levels of layout, the information from different levels of layout can be merged and mixed and this framework can be extended to more complex situations and different tasks.

Keywords: Document image categorization, Clustering, Non-negative Matrix Factorization, Script identification, Font identification, Language identification

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF ABREVIATIONS

DIU	Document image understanding
ED	Euclidean Distance
KLD	Kullbacl-Liebeller Divergence
NMF	Non-negative Matrix Factorization
PNMF	Projective Non-negative Matrix Factorization
NMTF	Non-negative Matrix Tri-Factorization
SVM	Support Vector Machine
KAS	K-Adjacent Segments
LDA	Linear Discriminant Analysis
SOM	Self Organizing Map
SIFT	Scale Invariant Feature Transform
SURF	Speeded Up Robust Features
SKL	Skeleton
BOW	Bag Of Words
MLP	Multi Layer Perceptron
ART	Adaptive Resonance Theory
LFDF	Light Field Distortion Feature
GMM	Gaussian Mixture Model
SGDD	Spatial Gradient Difference Descriptor

EDMS	Edge Direction Matrix
EDM	Edge Directional Map
BEMD	Bi-dimensional Empirical Mode Decomposition
IMF	Intrinsic Mode Function
LBP	Local Binary Pattern
GLCM	Gray Level Co-occurrence Matrix
TAS	Three Adjacent Segments
OFR	Optical Font Recognition
OCR	Optical Character Recognition
EM	Expectation Maximization

INTRODUCTION

0.1 Document image understanding

Invention of paper is one of the most important steps in the history of mankind and since the invention it is used in many situations. By the invention of the computer and dominance of the digital world, the electronic documents are substituting the paper-based documents by many reasons, such as easy sharing, easy searching or better preservation. Although, digital paper is the preferred mean in many situations, human used and still is using paper in many aspects of his life such as books, letters, mails and many others. Text recognition systems have been introduced and used in many areas such as posts, libraries and banks for automation. From another point of view, there are many sources of information such as ancient manuscripts and historical books available and many libraries own huge number of ancient manuscripts from different time-lines and periods. Many of these sources are converted to digital format (image) but in order to facilitate the usage, automatic document understanding methods are needed.

Different countries or regions use different scripts and there are countries that have more than one official script which are used in different regions or territories. The books are written in different scripts, font or writing styles and also different languages. Some scripts are shared between different languages and influences of different cultures and languages that have common script can be seen in different styles of handwriting and different fonts for printing.

In this multi-scripts, multi-fonts and multi-languages world , adapting one recognition engine to all of these variations is not efficient and a specific system is needed that selects the proper recognition engine based on the type of a document image. Different scripts use different symbols, might be written in different directions, use different shapes of the characters that might change their shapes or the connections between them according to the handwriting style (font) in a document image. Also, the interpretation of recognized characters or words is based on the language of that document image.

In this thesis, we focus on the task of document image categorization. Although, the main goal of document image understanding is obtaining textual information of a document image, the recognition systems need a lot of information for accurate results. Many of this information are related to the type of document image. In an example scenario for manual using of DIU system, many questions should be answered by the user in order to obtain the desired output. Such questions are, what is the script of the document image, what is the handwriting style of document image or what font is used to print that document image and what is the language of the document image. All of these answers have important and critical roles in automatic understanding of document image.

0.2 Writing system

A writing system or script is a tool to express concepts in a language and uses graphical shapes and symbols to represent elements and statements. Each script or writing system has a set of defined elements and each element is assigned with special meaning and role for writing in a language. Some writing systems have same origin but evolutions and changes of them during the time make them different. Writing systems of the world can be categorized into different groups. In Daniels & Bright (1996), writing systems, their features, differences and similarities are introduced and discussed. Writing systems can be divided into 6 main groups which is illustrated in Figure 0.1.

Logographic: The main example of this writing system is Chinese script in which each character (symbol) represents a single word or more.

Syllabic: The main example of this writing system is Japanese script and in this script a set of writing symbols represent syllables and words are made up of these syllables.

True Alphabetic: The main example of this writing system is Latin script and the elements of this system are called letters and each letter is roughly a phoneme of corresponding spoken languages.

Abjad: The main examples of this writing system are Arabic and Hebrew and each symbol is consonant. In most of cases the vowels (short vowels that are small symbols) are omitted and the reader should understand the proper vowels based on the context. These scripts are written from right to left.

Abugida: The main example of this writing system is Brahmic family of script in which the unit of the script is sequence of consonant-vowel (each unit consist of a consonant and a vowel).

Featural: The main example of this writing system is Korean script in which the symbols are the elements that make up the phoneme.

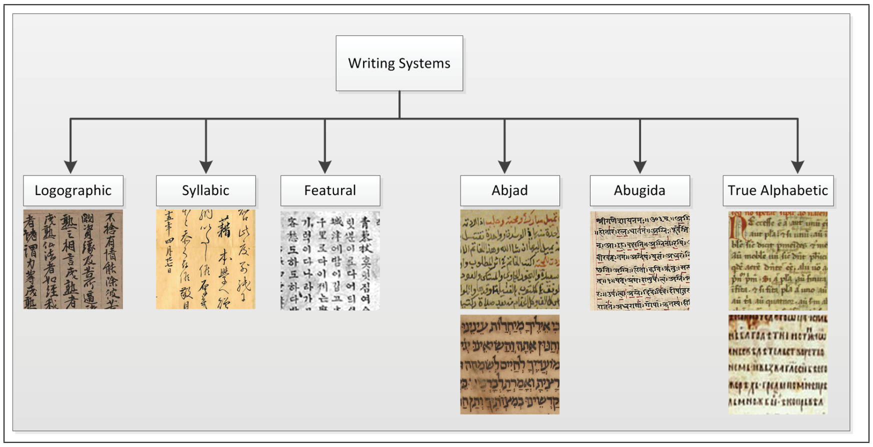
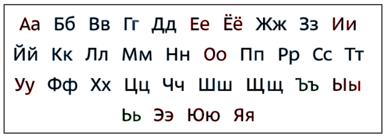


Figure 0.1   Writing systems and sample scripts

Document scripts

Latin script: Latin script is composed of 26 letters and some extra symbols are added in different languages that use Latin alphabets. The letters are written in isolated forms but the cursive style is invented by writing sequence of connected letters but the global shape of the letters generally do not change.

Cyrillic Script: This script is inspired from the capital Greek alphabets and introduced in east Europe. This script is the base of Slavic languages and Russian. The uppercase and lowercase of the letters are not much different compared to Latin script. The Cyrillic letters are shown in Figure 0.2.

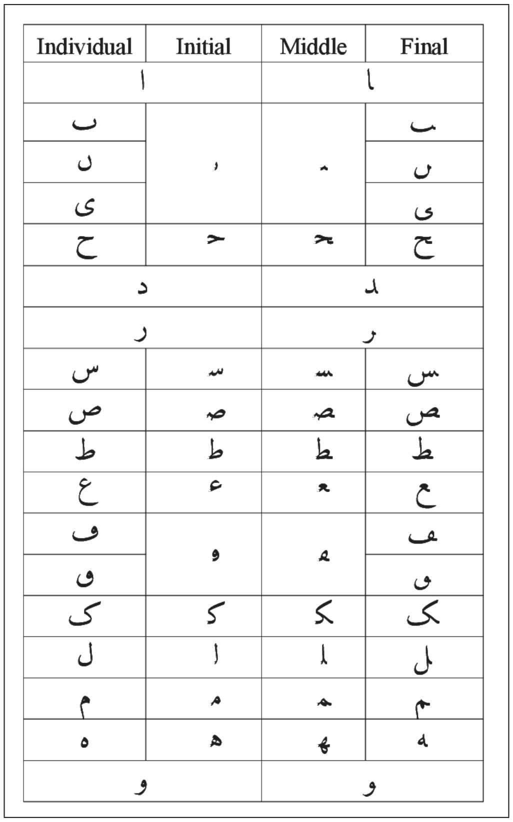


Аа Бб Вв Гг Дд Ее Ёё Жж Зз Ии
Йй Кк Лл Мм Нн Оо Пп Рр Сс Тт
Уу Фф Хх Цц Чч Шш Щщ Ъъ Ыы
Ьь Ээ Юю Яя

Figure 0.2 Cyrillic script letters

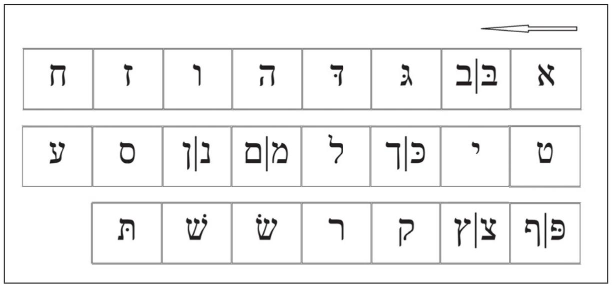
Arabic Script: Arabic script is composed 28 letters and it is shared between different languages therefor few other letters introduced to adapt it to different languages. This script is cursive and the letters attach to each other to create words. Each letter in this script can be written in two or four forms depending on the position in the word, the preceding and succeeding letters. Arabic letters and their different shapes are shown in Figure 0.3.

Hebrew Script: The Hebrew alphabet also known as block script is used to write Hebrew languages. This script belongs to the family of Abjad writing system and has 22 letters which 5 of these letters have different forms if they are used as the last letter of a word. Similar to Arabic, Hebrew is written from right to left but the letters are written in isolated forms. The present Hebrew which is also known as square script is derived from Aramic alphabets. The example of Hebrew script is shown in Figure 0.4.



Individual	Initial	Middle	Final
ا		ـا	
ٮ	ٮـ	ـٮـ	ـٮ
ں			ـں
ى			ـى
ح	حـ	ـحـ	ـح
د		ـد	
ر		ـر	
س	سـ	ـسـ	ـس
ص	صـ	ـصـ	ـص
ط	طـ	ـطـ	ـط
ع	عـ	ـعـ	ـع
ڡ	ڡـ	ـڡـ	ـڡ
ٯ			ـٯ
ک	کـ	ـکـ	ـک
ل	لـ	ـلـ	ـل
م	مـ	ـمـ	ـم
ه	هـ	ـهـ	ـه
و		ـو	

Figure 0.3 Different shapes of Letters (without dot) used in Arabic script



ח	ז	ו	ה	דּ	גּ	בּ|ב	א
ע	ס	נ|ן	מ|ם	ל	כּ|ך	י	ט
	תּ	שׂ	שׁ	ר	ק	צ|ץ	פּ|ף

Figure 0.4 Hebrew script letters

0.3 Calligraphy, Style and Font

Calligraphy is the art of writing and it can be found in all cultures. In some cultures calligraphy has a deep connection with paintings and its development is along with paintings. Some scripts such as Arabic is used in many countries and the influence of different cultures (that used

Arabic script) is seen in art of calligraphy. Calligraphy and writing style has a close connection in Arabic script. Many styles invented for different usages and many peoples improved the appearances of these styles from the artistic point of view. With the invention of printing many type sets and in digital form various fonts are designed and used. Although, these concepts (Calligraphy, Style and Font) are appeared in all of scripts, the focus of this thesis is on the fonts and styles related to Arabic Script. Some famous Arabic writing styles are as follows:

Koufi style: As one of the earliest writing styles used for writing many books, this style consist of mostly straight or angled parts and very few curves. The long and continuous horizontal and vertical strokes are the main features of this style.

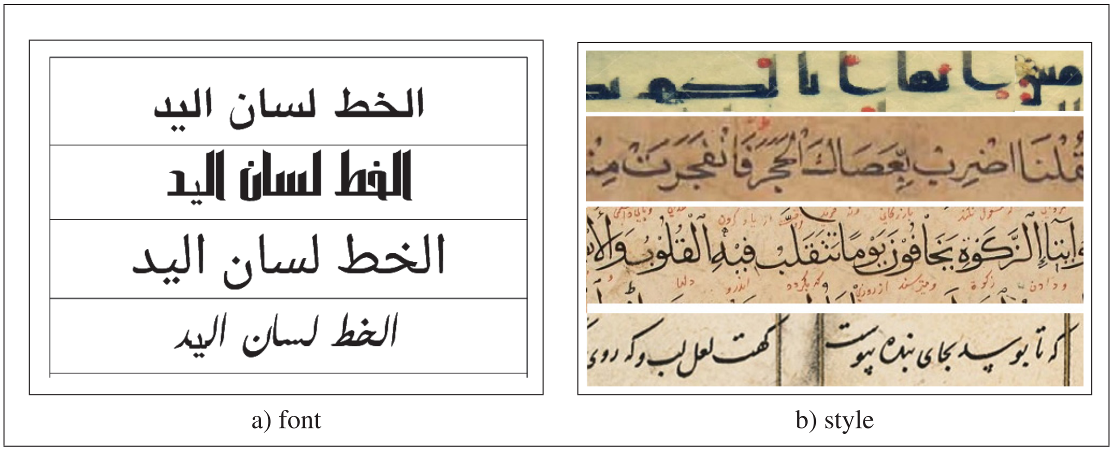
Naskh style: This style was introduced to remove the errors and ambiguities of Koufi style by adding dots and incorporating diacritics. The main features of Naskh are proportionality and scale which are considered in designing the letters. This writing style is one of the most legible and most used styles.

Tholoth style: This writing style is mostly used for artistic purposes and it seems to be similar to some other styles such as Naskh. This style has some distinctive features such as elongated forms for some letters or very shallow and open curves (circles). In this style some words are written with overlap (in artistic forms) which makes the reading of text a bit difficult.

Taliq style: This style is one of the first styles from Persian cultures. In this style the words are written with a lot of overlap and in different shapes but the writers should follow some basic rules. This style has many curves so lots of combination can be created to write with this style. One of the main features of this style is curved text lines.

Nastaliq style: This style is the most famous writing style in Persian culture, also known as Farsi style in some countries. This style is invented by combination of Naskh and Taliq and has many circles and the most of the letters are written based on curves.

In recent years, various fonts or type sets are designed for printing in Arabic script. One direction for designers was developing fonts with the aim of fast production without considering the details of the Arabic script and another direction was to design the fonts that follow the same rules and writing purposes of famous Arabic styles. Some examples of famous Arabic writing styles and Arabic fonts are shown in Figure 0.5.



a) font

b) style

Figure 0.5 Some examples of different a) Arabic fonts, b) Arabic styles

0.4 Language

Written language is a system of symbols and rules used by a group of humans for communication. Human invented script to preserve and transfer knowledge and then the scripts are adapted to the different languages. Arabic script is one of well known script and by the interaction between different cultures, it was selected by many nations and adapted to other languages such as Persian, Othoman-Turkish and etc. Although, there are many scripts that shared between different languages, the focus of this thesis is on languages that share Arabic script. Some example languages that use Arabic script are Arabic language(s), Persian (Farsi) language, Othoman-Turkish and Urdu. Although, these languages had and have many influences on each other, they have many differences.

0.5 Problem statement

Categorization of document image is a very challenging problem and these challenges are related to many sources. Some of the challenges are general and will be faced in any categorization systems and some others are related to the specific problem of categorization.

General challenges

Layout: Document images have very different types of layout that changes from simple to very complex structure. The complex layout is mostly appeared in handwritings or ancient manuscripts. One of the needed information before designing and using a categorization system is the level of the layout. Most of the proposed methods are based on the assumption that the layout analysis of the image is done with 100% accuracy and all of the models are built based on this assumption. From the layout analysis point of view, accurate layout analysis needs a lot of information such as the script which determines the constraints of creating document image such as writing direction or in-word space vs between words spaces.

Handwriting variation: Unless the book that written by trained writer whom they follow specific rules and specific style, the variation is high in the writings even between writings of an individual. It is not possible to find two writers with similar handwriting. Hand writing variability is very high between different persons writings but it is also significant (from the recognition point of view) between writings of of one person. More precisely, the handwriting of people differs on the following points: The spacing between words and the direction of the lines that gives the basic structure of the document. Most often the lines are expected to be horizontal, but they can have an increasing or a decreasing slope.

Noise and degradation: Due to aging and bad maintenance, ancient documents are often in degraded condition. The color change of the paper and diffusion of ink in the paper are the results of aging and disintegration of page, shears and stain as physical damage are the results of bad maintenance. In many situations, new degradation such as the bleed-through and deformations are also appeared after digitization process. Skew, scale and other distortions might be imposed to document image during digitization.

Pre-processing and feature extraction: Different types of features are extracted and used for document image categorization. Some features or methods are inspired from other fields. For-example texture features that are used for categorization, have no direct relation to the components of the document image so analyzing the results are very difficult. From another point of view, in order to prepare data for feature extraction and classification, many pre-processing steps are needed to remove unrelated variations. These steps highly depend on the layout and level of layout that should be extracted and also depends on the category.

Specific challenges

Challenges in script and language identification: The challenge for script identification is dealing with different fonts and styles. Other than handwriting variation that can be slight or hight variation for individuals, the font and style will change the nature of the document image hugely so the features should be robust against these variations and should be generalizable to other styles and fonts.

Challenges in language identification: The methods for language identification of document images are based on one idea, labeling components of the document image and then creating histogram from these labeled components. In text based methods, all complex steps of document image understanding (binarization, segmentation and recognition) should to be applied on an image to obtain these labels. In image based methods, the current coding procedure are not applicable to some scripts (especially cursive scripts) because of complexities in finding the components of those scripts.

Different nature of categories and related information: For any categorization task different levels of information is needed for accurate classification. For example, for script identification the assumption is that shapes of components are different for different languages and the features are extracted for this purpose. But if we consider font or style difference and other variations, this cannot be simply implemented and used for categorization. So at each level of categorization some variations are related to the classes and some are not so it is very difficult to design features that could be used for any task of document image categorization.

0.6 Contribution

Past research on categorization of document image established many methods for different categorization systems applied at different levels. The critical parts of such a system are representations and features that are informative, robust and relevant to the task of categorization. Many representations and features are proposed in literature and yet most of them cannot be used in real applications and practical situations. Therefore, **the main goal of this thesis is to propose a framework for efficient document image categorization**. This will be done by designing novel methods for script, style and language identification in a framework that can also be adapted to more complex problems.

First, this research focuses on the problem of script identification of ancient manuscripts which differentiates writing systems using the elements of scripts. Current methods are proposed mostly for machine-printed document image and many of them ignore the components of the scripts (in global methods) or ignore redundancy of information for extracting the correct information (in local methods). The efficient representation and feature extraction framework is proposed based on patch representation and automatic feature learning. It will be shown in the experimental section that the proposed method is robust and flexible and improve the performance of categorization.

Secondly, this research will focus on the problem of font and style identification of Arabic script. Many features are proposed for this task but the focus of current methods is toward very constrained situation where the level of identification is known and fixed. The global features ignore the local properties of the fonts and the local features are not robust and can only be applied on few classes of the scripts. The efficient patch-based representation with automatic feature extraction is proposed to solve this problem in a broader levels of layout. This representation and the corresponding features are evaluated in experimental parts to show their robustness and performance for the problem of style identification.

Finally, this research will focus on the problem of language identification in Arabic script languages. The main idea of language identification is to assign labels to the components of the script and then obtain a representation that shows the frequency of occurrences of those components. This process is done through two approaches, a recognition system with high accuracy or a recognition with consistent error (not random) and a coding scheme that assigns group labels to image components and compare these codings with textual data. The first approach needs application of many complex steps toward recognition and the second approach is not applicable on some scripts such as cursive script. The effective patch-based representation is adapted to this problem by proposing a novel method for language identification of Arabic script languages. In the experiments, this method is compared to the n-gram features extracted from textual data and the results show the strength and performance of the proposed method.

This thesis will provide a framework for representation and feature extraction of document image toward the categorization problem (initially for Script, Style and Language) that can be adapted to more complex problems such as content-based document image categorization.

0.7 Context of the thesis

This section details the scientific context of this thesis. The methodologies for document image categorization systems are inspired from several fields such as computer vision. During our research, we were interested in proper representation that could be used for different categorization applications and a set of features that can be extracted from document image in an automatic way and used for different tasks.

One main area that limits the capability and applicability of current methods is representation. Most of categorization methods are proposed with an assumption about the specific level of layout of the target image so the steps of algorithms such the representation and feature extraction will be limited based on that assumption.

Different scripts and writing systems have different elements and incorporate different constraints and rules for writing. Some scripts are composed of isolated letters and there are no connections between letters and in cursive style of these scripts, writing the connected sequence of letters, the shapes of the letters do not globally change. Some scripts are cursive in nature and the letters should be attached to create words and the shape of the letter should be altered according to the position in the word and also the preceding and succeeding letters. In some script the notion of character or letter does not exist and each component is a word or more. In some script the writing direction is left to right while in other scripts the writing direction is right to left or top to bottom.

One of the most efficient way of representing the components of different scripts in an unconstrained layout is using image patches. Patch-based image representation is used in many applications and strength and performance of this representation is confirmed by many experiments in many different situations. The patch representation gives us the flexibility of skipping the complex step of lyout analysis and also the potential to overcome many limitations that are imposed on current methods. Image patches are robust against noise, degradation and discontinuity in textual objects and can be extracted efficiently.

Some of successful methods for object retrieval are based on keypoints detector and descriptor and these methods combined with BOW model are used successfully in many applications. Unlike natural image that the position of the objects and location of keypoints are unknown, in document image we roughly have the position of keypoints i.e. foreground pixels that can be obtained by binarization methods. From another point of view, document images contain highly redundant information and this information can be used for extracting the correct information. Unlike the keypoint detectors that mostly work based on corner detection and can estimate the scale of the corner ponits, we use global properties of document image to determine the patch size. The global properties of document image such as average text height, average stroke-width and void space between text lines are used for this purpose.

The next step of the process is feature extraction, obtaining an abstract descriptor that should be robust to unwanted variations. For the current methods, the features or descriptors are designed based on few observations so they might not reflect the true nature of data. Here, we are interested in automatic feature extraction methods where the features are learned from the variations in data and the redundancy in data is exploited for obtaining robustness and generalization power.

The aim of Non-Negative Matrix Factorization is factorizing data matrix as a product of two matrices with the non-negativity constraint. This constraint is observed in many real data and also have physical interpretation that shows the presence or absence of objects as the components of more complex objects. These two matrices are interpreted as a basis matrix and a coefficient matrix which the basis matrix contains the building blocks of the objects and coefficient matrix contains the contribution of those building blocks. Two properties of NMF are low rank approximation and collaborative learning that means the number of bases are less than dimensionality of data so noise can be eliminated and collaborative learning helps to retrieve the correct and accurate information.

Although, the above strategy seems very promising for script identification, it is not applicable for the style or font identification in a specific script. As it is observed in the previous figures, shape of letters in different styles have global similarity but they are different in very fine details. The motivation for style identification is that if the bases that obtained from the other styles are used to represent the patches of specific fonts, the error of representation will be higher compared to the situation if the correct bases are used. Although, NMF seems very suitable for this application the bases that obtained from data of different fonts are very local and could be similar but in different order. NMTF as the generalization of NMF to three matrices can be used here. The motivation for this method is that tri-factorization method extract bases from patches of different font and then use these bases to recreate some elements of the fonts (as cluster centers). As this combination is different for each font, the bases are specializes toward that font and these bases will be more discriminant.

Although, some languages use same script, the structures of the languages and their roots are different so the elements of the scripts are used differently. If the textual information of document image is available, the simple way for language identification is creating histogram of the objects (n-gram features) and then useing them for this purpose. In the absence of textual information, the most effective model is bag of visual words in which the aim is to find some common objects between differnt classes and then use a representation based on those objects for classification. Image patches can be considered roughly as equivalent of textual objects (i.e. letters if selected correctly) and most of the concepts for textual data can be transferred to image domain by patch representation.

In BOW approach, image will be modeled by a histogram that shows the frequency of appearance of some objects or words in that image. The main part of this approuch is obtaining this set of words or keywords that is called dictionary learning. Most of keypoints or keywords detector are based on finding all of the words from the data and using clustering to create keywords or dictionary. This dictionary will be used for representation of elements of document image. Unlike natural images that can have infinite number of words, document images are composed of few objects and while the shapes of these objects have variation (due to being handwritten or having distortion or different font), they are similar at the abstract level .

Simple clustering methods are proposed to cluster samples or features but bi-clustering method are proposed to cluster data simultaneously so these methods are able to find the interrelation between samples and features and they provide better clustering results. Non-Negative Matrix Tri-Factorization is inspired from the relation of NMF and clustering methods and proposed to exploit this relation. In a very simple interpretation, NMTF is projecting data to a low-dimensional space and provides the abstract representation for data and then explores that space to find relevant clusters. The cluster centers or keywords obtained by NMTF are more accurate and better representative compared to simple clustering method such as kmeans.

0.8 Outline of the thesis

The introduction chapter contains the general context of the thesis. Chapter 1 contains the reviewed state of the art methods with the highlights on limitation and drawbacks of the current methods. In Chapter 2 the general methodology with the objectives of the thesis are mentioned. The framework for script identification, style (font) identification and language identification are then described. The next three chapters present the methods proposed and developed in this thesis and the corresponding results. Chapter 3 presents the first journal article, script identification method. Chapter 4 presents the second journal article, a style identification method. Chapter 5 presents the third journal article, a language identification method. Then, Chapter 6 provides the general discussion that highlights the improvements that achieved with the proposed methods and discuss the drawbacks. Finally, the general conclusion summarizes the work presented in this thesis with a glimpse of the future perspectives.

LITERATURE REVIEW

In this chapter we briefly describe state of the art methods for script, style(font) and language identification.

As mentioned before, script is defined as the graphic form of the writing system used to write statements that refers to a particular way of writing and the set of characters used in it. A script may be used by only one language or may be shared by many languages, sometimes with slight variations from one language to other. Some languages even use different scripts at different points of time and space. Therefore, in this multilingual and multi-script world, OCR systems need to be capable of recognizing characters irrespective of the script in which they are written. Manual identification of the documents scripts may be tedious and time-consuming. Therefore, automatic script recognition techniques are necessary to identify the script in the input document and then redirect it to the appropriate character recognition module. A script recognizer is also useful in reading multi-script documents in which different paragraphs, text blocks, text lines, or words in a page are written in different scripts. For scripts used by only one language, script identification itself accomplishes language identification. For scripts shared by many languages, script recognition acts as the first level of classification, followed by language identification within the script.

1.1 Script identification methods

Different script can be separated by means of specific spatial distribution of characters and visual attributes which differentiate them. The first step of script identification is extracting useful features from a set of document images and the second step is using these features to classify the document images. These approaches may be divided into two categories based on the type of the features:

- Structure-based methods:

  Structure, connection and writing style of different scripts are good features to differentiate between scripts. In most of these methods, connected components (continuous runs of pixels) are analyzed based on their shapes, structure and morphological characteristics to produce discriminant features.

- Visual appearance-based methods:

  Difference of the shape of individual character, grouping of character into word and grouping of word into sentences can be tracked without really analyzing the pattern of characters in the documents. So, several features are proposed to describe the visual appearance of script in a region.

Based on the level of applying a method inside document, the script identification methods may be classified into different groups: a) page wise b) paragraph wise-block wise c) text line wise d) word wise

Structure-based method

In Pal & Chaudhuri (2001) the script identification method for printed English, Chinese, Arabic, Devnagari and Bangla text lines is proposed. To separate Devnagari and Bangla from the other script they used the headline features that show the long horizontal line in horizontal projection profile. The position of this longest run is used to separate Arabic from Devnagari and 6 Bangla. They use zone-wise features to separate Bangla and Devnagari and number of vertical run to separate Chinese from Arabic and English.

In Ablavsky & Stevens (2003) a script identification method to separate degraded printed Cyrillic and Latin is proposed based on subsets. After pre-processing and binarization, the Cartesian moments features are extracted from connected components and normalized to obtain scale invariant features. Hu moments are used to compensate for changes in the rotation. The other features such as curvature, holes, the length of the major and minor axis, eccentricity, elongation convexity and co-occurrence features are also extracted and the subset selection algorithm of RELIEF-F is used to find the most discriminant features.

In Pal *et al.* (2003) the authors proposed a hierarchical script identification method for Indian Documents. Their features are headlines, horizontal projection profile, water reservoir principle based features, left and right profiles and features based on jump discontinuity.

In Shijian & Chew Lim (2007) a component vector of size 10 that contains number of vertical cut for each connected component is used for script identification. The components of 1 to 8 of these vectors are number of vertical cut of connected component. The 9th and 10th elements are constructed by counting number of vertical cut lying above and below the connected 7 component center. The cosine similarity is used to measure the angle between query and reference component vector.

In Hangarge & Dhandra (2008) two sets of global and local features and decision tree strategy are used for script identification. The global features are stroke density, pixel density and local features are aspect ratio, eccentricity, extent and directional profile densities.

In Chanda *et al.* (2009) the word-wise script identification method for 3 classes of Sinhala, Tamil and English texts is proposed using structural, topological and water reservoir principle features. They proposed a set of 8 features that include right convexity feature, bottom, left and right reservoir based feature, size dissimilarity of loop, position and size of dots, top reservoir and distribution of vertical stroke feature. Given a training dataset, RBF kernelled SVM classifier is used to classify new image.

In Padma & Vijaya (2009) the text-line-wise script identification method for Telugu, Hindi and English scripts is proposed. In their method, after pre-processing and top and bottom projection profile, a set of features are extracted and analyzed based on the script to find the ranges of the features in mentioned scripts. To classify new image, this set of features is extracted and the script is classified based on the ranges that those features lie. The rejection strategy is used to separate scripts other than mentioned scripts.

In Rezaee *et al.* (2009)(Rezaee, Geravanchizadeh et Razzazi, 2009), the authors proposed a method for script identification of machine printed Persian and English. After skew correction and segmentation to text line and words, the distribution of horizontal projection for each script is calculated. According to their observation, the Latin and Persian words or text lines have different distribution while the former has almost uniform distribution, the latter has normal distribution.

In Aithal *et al.* (2010) the authors proposed a Textline-wise trilingual script identification method based on horizontal projection profile for separating Kannada, Hindi and English scripts. The feature that they used is based on the first and second maxima of horizontal projection. They proposed the ruled based classifiers that separate the mentioned scripts in 2 steps. In the first step the first maximum is compared to 1.5 times of the second maximum to separate Hindi from others. In the next step they calculated the mean of projection profile between first and second maximum and find the value of point after maximum in horizontal projection profile. Comparing these two values and defining the ranges for features, Kannada and English can be recognized from each other.

In Chanda *et al.* (2010) the authors proposed the method for script identification method for Chinese, Japanese and Korean script. They used chain-code histogram-based features inspired from the observation that the difference of Han based scripts such as Chinese, Japanese and Korean have the dissimilarities that can only be found in small part of the characters. In their method after computing the bounding box and dividing it into some blocks, each block is described by a vector of size 4 that contain frequency of chain code. The feature vector for each character in bounding box is concatenation of all features obtained from correspond blocks. These features are normalized by dividing by the maximum value and dimension is reduced by down sampling with Gaussian filters. The classifier that they used is SVM with Gaussian kernel.

In Gopakumar *et al.* (2010) the zone-based script identification method is proposed to separate South-Indian, English and Hindi script. For each text line, 3 groups of features are extracted in 3 steps. At first, 9 features are extracted from text line and then the text line is divided into 3 horizontal zone and for each zone the set 9 of features are extracted and for each text-line

Euler number is computed. To increase the classification rate, the most prominent features are selected from the whole set of 32 features. SVM and K-NN classifiers are used for the classification of new document images.

In Roy *et al.* (2010) the script identification method at word level for separating printed Kannada and Devnagari documents mixed with printed/handwritten English numeral is proposed. The main steps are, pre-processing and layout analysis for line-wise and word-wise segmentation using horizontal and vertical projection. A set of features form connected components are obtained by concatenation some features such as average aspect ratio, distribution of pixel around the center of mass. They use K-NN to classifying new features. 10

In Khoddami & Behrad (2010) the script identification method for separating Farsi and Latin script is proposed based on curvature scale space. Each connected component is represented as a planar curve and evolved by convolving with 1-D Gaussian Kernel iteratively. At each iteration, curvature of the evolved curve is calculated and a histogram of positions that curvature becomes zero versus the variance of the Gaussian Kernel is depicted until there is no zero crossing. The histogram is renormalized to become robust against scale, noise and orientation change. Their features are set of 10 measurements from the difference and ratios and a set of extra 5 features to deal with round shapes. K-NN Classifier with hierarchical distance measure is used to classify new connected component.

In Marinai *et al.* (2010) the authors proposed a script identification method based on bag of characters and SOM for separating Cyrillic, Latin and Greek scripts. The main steps of learning phase are extraction of symbols/characters, representation with feature vectors, vector quantization by applying clustering, representation of symbols with index of cluster and representing a document image with weighted frequencies of symbols. For each symbol (connected component) a feature vector that describes the distribution of pixels is calculated and Self Organizing Map (SOM) is used for clustering and vector quantization. For new Image, the symbols are extracted and indexed by SOM and the weights corresponding to cluster for pages are calculated. The similarity of query and indexed page is the cosine of the angle between weight vectors.

Visual appearance-based method

In Singhal *et al.* (2003), the authors proposed a method for script identification of handwritten text documents. After pre-processing steps such as denoising, thinning, pruning, m- connectivity and text normalization, they applied multi-channel Gabor filter to extract the texture features to separate different scripts using visual appearance characteristics of document images.

In Busch *et al.* (2005), the authors proposed a method for block-wise script identification. The texture features of gray-level co-occurrence matrix, Gabor energy features, wavelet energy features, wavelet log mean deviation features, wavelet co-occurrence signatures and wavelet scale co-occurrence signatures are used to separate Latin, Chinese, Japanese, Greek, Hebrew, Sanskrit and Persian. For each co-occurrence matrix, 8 features of energy, Entropy, Inertia, Contrast, Local Homogeneity, Cluster shade, Cluster Prominence and infimum Measure of correlation are extracted and LDA algorithm is used to reduce the dimensionality of data. The GMM that models the distribution of each class by combination different Gaussian distributions is used to perform the classification.

In Pan *et al.* (2005), a block-wised script identification method is proposed based on steerable Gabor filters. They developed rotation invariant features that obtained from mean and standard deviation of filtered image using 1-D Fourier transform. To speed up the feature extraction step, they used the approximation method that instead of calculating the response of the filter in all direction, use steerable properties to combine the filter outputs that correspond to the basis of the rotated version filter. For the classification, they use two-layer feed forward back propagation neural network with rejecting strategy by 4 neurons in the output layer according to four classes of scripts.

In Ben Moussa *et al.* (2008), the script identification method based on fractal feature is proposed for printed/handwritten Arabic/Latin scripts. Fractal dimension is a measure of 12 complexity of an image and can be estimated by the fraction of the logarithm of number of boxes by size of those covers object, to the logarithm of inverse of size r. To gain the good result they use multidimensional b-features and d-features of full image, horizontal and vertical projection profiles. The b-features are based on box counting and d-features are obtained by texture surface measurement after applying dilation operator in different levels. They used K-NN and RBF to separate different classes.

In Linlin & Chew Lim (2008), the authors proposed a method for script identification of document images under the perspective distortion. They approximated the perspective distortion by affine transform and for each character generated a signature that can be captured correctly without knowing information about the text-line direction. To create templates they used a method to extract the signatures and clustered them by hierarchical clustering algorithm. Using cosine distance, the 30 biggest clusters are chosen and centers of these clusters are considered as templates. Histogram of nearest template is used to differentiate between scripts.

In Chanda *et al.* (2009)(Chanda et al., 2009), the authors proposed a word-wise script identification method for separating Latin, Devnagari and Bangla in two stages. The first step performs high speed identifying and the second step deals with samples that are degraded or have low recognition confidence in the first step. In the first step 64-dimensional chain-code histograms of bounding box that is divided into few blocks are computed. In the second step a feature vector of dimension 400 is obtained by dividing the normalized gray-level image -mean filtered of binarized image- into several blocks, applying Sobel filter, calculating quantized angle and the magnitude of the gradient. The classifier that they used is SVM with Gaussian Kernel.

In Hiremath *et al.* (2010a), the texture-based script identification for seven Indian languages and Latin script is proposed. They used Har wavelet to decompose any document image into 4 sub bands. For each pair of (A, H), (A, V), (A, D) and (A, abs(V-H-D)) , two co-occurrence 13 histogram in 8 directions are calculated to construct the cumulative histogram and to calculate 3 features of the slope of regression line, mean and mean deviation. The procedure is repeated for the complementary image obtained by . They used the Euclidean distance and K- NN classifier to classify the new image.

In Zhou *et al.* (2010), the authors proposed a texture-based feature extraction for script identification of six scripts. After wavelet decomposition and calculating energy feature using three detail sub-bands, linear quantization of the features is used to prepare features for calculating wavelet energy histogram moment features. At next step, wavelet energy histogram is calculated and four weighted moments features –summing over first n value of original, low-pass, band-pass and high-pass filtered histogram- are calculated from each histogram. SVM is trained to classify new features.

In Yuemei *et al.* (2011), the script identification method based on global and local texture features is proposed. In the first step the image is decomposed into different intrinsic mode function (IMF) by Bidimensional Empirical mode decomposition (BEMD). Local binary pattern (LBP) algorithm is used to detect the local features of the textures and SVM is trained to classify new image.

Despite the fact that there are several styles that invented for writing Arabic script family, there are a few methods that addressed the identification of handwriting Arabic style. Demanding of high performance OCR system causes the evolution of document image categorization algorithm to select the proper OCR system that adapted for specific script. One of these systems is font identification system that classifies new document image based on the type of the font that used to print that document.

1.2 Style identification methods

Font identification methods can be divided into two categories based on type of method:

- Methods based on global features, these methods are text independent and skip information about the letters and the features describe global properties of document image.
- Methods based on local features, these methods are based on analyzing the individual object (letters). In these methods the components of document image (letters) are isolated and then based on similarity from textual point of view they differentiate between different fonts.

The font identification methods can also be classified into 4 groups, based on type of features:

- texture features
- typographical features
- structural features
- automatic learning of features

In Yong *et al.* (2001) the authors proposed a font identification method based on texture features. As feature extractor step, they used Gabor filter in 4 different orientations and 4 different scales that produce 16 Gabor channels. The mean values and standard deviation of output channels are selected as texture features. The similarity measure is Euclidean distance that weighted by standard deviation of features of specific font.

In Fang *et al.* (2002), the texture-based font identification method is proposed. They used Gabor filter as feature extractor and GA algorithm to optimize orientation set for Gabor filter. The range of orientation of 0 to 180 degrees with steps of 5 and 15 degrees is used for the input of GA and the fitness function that designed is: and are parameters they can be set based on experiments and unknown font feature is classified by Euclidean distance.

In Li Zhang (2004), the authors proposed an italic font recognition method based on wavelet decomposition. After applying wavelet transform, horizontal, vertical and diagonal sub-bands are analyzed and stroke patterns that differentiate italic and regular fonts are found by experiment. These rules are used to separate Regular and Italic Fonts.

In Hung-Ming (2006), the font identification method based on stroke template matching is proposed. After thinning process that creates skeleton, endpoints are selected and small branches are discarded. Then the skeleton image is scanned from left to right and top to bottom and stroke 15 junction with the length of more than defined threshold and strokes reaching endpoints are kept as templates. In the recognition phase the extracted templates are compared with templates in the training set and Bayes's decision rule determines the best matching.

In Ding *et al.* (2007), the authors proposed the font identification method for single Chinese characters. They applied a 3-level wavelet transform on the image of single character that normalized to size 48*48 pixels. For each sub-band, they divided it into some non-overlapped blocks and for each block they calculated a feature that is the weighted sum of wavelet coefficients by Gaussian kernel with two parameters of that should be set based on size of blocks. The result is one M*M feature matrix for each sub-band. The feature vector of size 702 is created for each character image and LDA is used to reduce dimension and create discriminant feature set. They used MQDF (modified quadratic discriminant function) classifiers and features of one prototype for each font class, to classify new inputs.

In Jamjuntr & Dejdumrong (2009) the authors proposed a font identification method based on linear interpolation analysis. The first step of the proposed method is detecting the contour points of each character and indexing them by the edge detection algorithm. The contour pixels are sampled by the same amount for all characters and linear interpolation is created from these sampling pixels. The angles between every two pairs of obtained lines are calculated. Each new character is classified by the minimum Euclidean distance between the obtained angles.

In Ramanathan *et al.* (2009a) the authors proposed a font identification method based on Gabor filter and SVM. They applied Gabor filter on normalized image that divided into 9 non-overlapping blocks and 50 features (mean and standard deviation) are extracted for each block. SVM classifier is used to analyze new data. In Slimane *et al.* (2010) the authors proposed a font recognition method for Arabic script. In their method, each normalized word image that is kept in gray level is scanned with sliding window and in each position of the window a set of 10 features is extracted and the feature vector for each word is created by concatenation all features obtained in previous step. These features are number of black components, number of white components and the ratio of black/ white component and etc. and etc and likelihood of the features are estimated by GMM.

In Azmi *et al.* (2011) the new technique for font identification of handwritten document is proposed based on scalene triangle features. The pre-processing step is segmenting Jawi alphabets into initial, middle, end and isolated state. In feature selection step 3 important points; first black pixel on the right, first black pixel on the left and centroid of images are selected and labeled. 21 features are extracted from the triangle formed by 3 points and used for classification.

In Bataineh *et al.* (2011) the authors proposed a font identification method based on edge direction matrix (EDMS). After normalization of document images that includes skew correction, text-line extraction, lines normalization and text block normalization, Laplacian filter is applied to find the edges of texts in the image block. EDSM that is a statistical feature extractor based on texture analysis is applied to extract the features. First order and second order EDM are extracted from the edge image. In the recognition step they use back-propagation neural network with 22 node for input, 18 node for hidden layers and 7 nodes for output layer.

In Pourasad *et al.* (2011) the authors used spatial matching algorithm for font identification of Arabic script. In their method, contour points of individual Arabic alphabets are extracted and Euclidean distances are calculated and quantized logarithmically to reduce the computation time. Respective slope between two points are calculated and quantized into 12 intervals. For each contour point the (SGDD) descriptor vector of size 60 is created and this SGDD is used to recognize the type of fonts.

Language identification is the problem of automatically determining predefined language of a passage of text that is written in a document. Effective information retrieval by a user is difficult without correct information of language. It is an important step for a variety of language processing to identify the correct language for use of readers. Language identification is needed in text-based documents for effective information retrieval. Without the basic knowledge of the language the document is written in, applications such as information retrieval and text mining are not able to accurately process the data, potentially leading to a loss of critical information. The problem of written language identification is attempted to be solved for a long time and various features based models were developed for written language identification.

1.3 Language identification methods

Language identification methods can be divided into two categories based on type of data:

- Text-based methods where the textual information of document image is available. In these methods, the n-gram representation or keyword representation followed by some other techniques -to reduce the influence of the error- is used for language identification.
- Image-based methods where the component of the document image is coded by some measures and few features are extracted based on this coding and the relation with textual information and these features are used for document image representation and language identification.

Text-based methods are based on the assumption that the textual information of the document is available, either the document is digital born or the textual information is obtained by OCR/recognition system. For the digital born documents the textual information are accurate and the n-gram features or the frequent words as histogram or textual data processed by more advanced method is used for classification. The OCR based methods are based on the assumption that the accuracy of the OCR engine is high or at least the error produced by OCR is consistent (not random error) so the proposed methods use the similar features or representations for language identification. One note for these type of methods is that all of the images should be processed by one OCR engine and this engine should be changed, altered or modified.

Image based methods -which are mostly developed for Latin-based languages- are based on the assumption that the component (individual letters) of a document image can be separated from the rest of the letters in a word. These components are codded (based on some observations) by defining some features such as ascenders, descenders and few other distinctive features for grouping the components into few categories. Then the sequence of letters (words) is codded by these categories and this coding is compared to the textual information to find the most frequent words and their corresponding codings. Then, the language of a document image can be identified by comparing this information to the set of labeled information.

In Shijian & Chew Lim (2007) the authors proposed Latin-based language identification method for document images. After text-line and words segmentation, horizontal projection profile is calculated and 2 peaks are labeled as x and baseline. The strokes are classified into 8 categories and individual characters or combinations of characters are coded based on category or categories that they belong. Then, most frequent words associated to each language are extracted and coded based on 8 categories. The histogram of shape code and frequency is used to identify language of new document images.

In Selamat *et al.* (2007) the authors are focusing on Persian letters for decision tree approach. Intuitively, common words such as determiners, conjunctions and prepositions are good clues for identifying languages. They apply the decision tree approach to identify the most apparently suitable language for each significant letter in the given document. Since only the letter context is used, and no frequency information is stored in the tree, a very simple presentation is obtained. The tree is composed of a root node from original text and leaves which are the decision tags.

In Ljubešić *et al.* (2007) that proposed for Croatian, Serbian, and Slovenian language. Firstly, the authors extracted the list of most frequent words from the Croatian corpus. They measured the frequency distribution of the documents in the test corpus (4364 documents in each of 3 languages) regarding the percentage of N most frequent Croatian words each document contains. They used these distributions for the categorization of new documents.

In Selamat & Ng Choon (2008) the authors assume language identification task as a text categorization problem. It involved both training and testing process using machine learning methods. Training is the process for the language identifier to learn the pre-defined input patterns 18 produced by feature selection. Then, the well-trained identifier can be used to predict the unknown documents into the predefined language categories.

1.4 Discussion

Researchers have attempted to characterize different scripts, fonts and languages either by extracting their structural features or by deriving some visual attributes. Accordingly, many different features have been proposed over the years for categorization at different levels within a document such as page wise, paragraph-wise, text line-wise, word-wise, and even character-wise. Text line-wise and word-wise categorization systems are particularly important for use in a multi-category document.

However, compared to the large amount of literature available in the field of document analysis and optical character recognition, the volume of work on categorization is so small. The main reason is that most research in the area of OCR has been conducted to solve issues within the scope of the country. It is noted that most of these identification methods have been tested on machine-printed documents only and only for few scripts, and their performance on handwritten documents is not known. In view of this, it will not be wrong to say that categorization in handwritten documents is still in its early stage of research.

In this section we discuss the limitation of current state-of-the-art methods for document image categorization with respect to script, style and language.

Limitation 1: Fixed level of layout

The first limitation of the methods for script and style identification is the assumption about the layout the image. That means all of these methods are proposed based on the assumption that the layout and the level of layout that the method would be applied is known and fixed. So if the method is proposed for paragraph based identification, it cannot be used for other levels of the layout. The second is all of the features are designed based on the level of the layout and the features are fixed to that level and cannot be generalized to other levels. The other drawback is that all of the methods are based on completely accurate layout analysis (mostly manual layout segmentation or synthetic data) therefore these type of methods are not practical.

Limitation 2: Feature design and extraction

The second limitation of methods is that the features are designed by experts (user) and in an ideal situation. Therefore, these type of features have three aspects, discriminant property, generalization and extraction accuracy. These features are designed by observation of few classes and may not consider all of the situation so they might not be discriminative if applied on new document image. The second concern is generalization and these features might not have generalization power. The third concern is implementation, how these features can be extracted accurately. Document images are the subject of many situations like noise degradation, transformation and etc. so these features the extraction way needs many adaptations and many considerations.

Limitation 3: intensive normalization

Most of the method for document image categorization based on texture concept needs many pre-processing to normalize data and remove unwanted variation in data. These methods need normalization of text height, stroke width, and for transformed based features need specific size for processing so sometimes data replication is needed. In most of the cases these pro-processing steps are class dependent and the class determines the parameters of pre-processing.

Limitation 4: Limited Scope

In most of methods the features cannot be generalized to other classes. For example, local based features that are extracted from components of the Latin script cannot be generalized to Arabic because of the difference between the nature of those scripts. Language identification method proposed for image-based identification Latin languages are not applicable to Arabic script and also cursive style of Latin due to difference between these scripts.

Limitation 5: Global features vs local features

Global features are robust against noise and degradation but they need large portion of data. Local methods are fast and can be applied to smaller parts of the text but these features are very sensitive to noise and degradation.

CHAPTER 2

GENERAL METHODOLOGY

In this chapter we briefly explain our general methodology and the motivations for this work. Also, we briefly explain the the general methodology which will be discussed with details in the following chapters. In this thesis, we focused on 3 problems of script, style and language identification which is according to hierarchical nature of these categories and top-down problem. In script identification step the shape of the building blocks of the script are different. In style identification and the related field font identification, the fine details of the component determines the different categories and in language identification the usage of the elements are different. There is natural connection between these three steps and as it can be seen from the methodologies, we used similar strategy but different methods which are adapted to each step to tackle these problems.

A successful framework for categorization should have two properties, proper representation and robust features. The representation highlights and reviles the relevant information that should be extracted and the features provide a robust and an abstract description for the objects. Accuracy of the classification methods highly depends on the quality of these two steps. The proposed methods for categorization are shaping a framework for representation of document image and feature extraction that can be adapted to more complex tasks by adding constraints to the objective function.

2.1 Research Objectives

As mentioned in the introduction section, the **main goal of the thesis is to investigate the problem of document image categorization**. It will be addressed in 3 specific objectives related to script, style (font) and language identification.

2.1.1 Objective 1: to propose a new method for robust script identification of ancient manuscripts

Existing method for script identification mainly are proposed for machine-printed document image and most of them are based on the specific and fixed layout. From another point of view, the features are designed based on the limited classes of scripts that might not be accurate if new script is added. So the first objective of this research is to propose a new method for script identification of ancient manuscripts with the aim of having flexible and robust method that can also be generalized to any type of script. Two main parts of this system are proper representation and accurate feature. A good representation reveals information and discriminant features provide the compact descriptor for accurate classification. Our motivation for script identification is that the elements of different scripts are discriminant enough and can be used for script identification. These elements are not used with specific shapes all the time and their shape might change due to different handwriting, styles or fonts. Also some scripts have cursive nature so the elements change their shapes in a word according to their positions. Therefore, these elements cannot be obtained by simply analyzing the components of a document image. Patch-based representation of images finds lots of attention and used in many applications. The patches can be extracted easily and if the positions and the sizes are correct, they can be parts of script elements, an element or combination of some elements. Then these patches can be used for processing and classification. Although, these patches contain useful information, there are some limitations related to them such as high dimensionality and variation in shapes of the objects. The best way to describe data is to let the algorithms explore data and observe variations in order to find an abstract representation of data which is also free of unnecessary information. Non-negativity constraints on data and presence of noise and degradation lead us to NMF, that provide the low rank approximation of data by collaborative learning and non-negativity constraints. Our proposed approach which is based on patch representation and NMF will be discussed in Chapter 3. It provides the first method for script identification based on patch representation and automatic feature learning.

2.1.2 Objective 2: to introduce a new method for flexible style and font identification of document images

Existing method for font or style identification are proposed with a process in which an expert observe data (and based on experiments) then defines a representation and introduces some features that describe the differences in fonts. Due to lots of variation in the layout of document image, the level of identification will be fixed from the beginning and the representation and features obtained form a specific level of layout might not be used for other levels of the layout. Global based methods (mostly texture based methods) needs a big part of an image (paragraph) and lots of pre-processing steps for removing the unrelated variations. The local methods need the objects (letters of the scripts) to extract features and are sensitive to noise and degradation. The patch-based representation and automatic feature extraction provide a good framework for this problem. Patches contain the objects or part of an object and by using automatic feature learning, with collaborative learning, the variation related to the font and styles will be captured in few bases that can be used for classification. Therefore, the second objective of this research is to provide a new method for font identification of machine-printed document that can be generalizable to handwritten manuscripts. This method will be explained in Chapter 4. The contribution of this approach is a flexible and accurate patch based method with automatic feature learning for font and style identification.

2.1.3 Objective 3: to propose a new method for accurate language identification of printed/ancient manuscripts

Although, some languages use the same script and almost the same elements (characters) for writing, they are different in that way that these objects are used as sequences to construct words and sentences. This inspires many researches in the field of document understanding to introduce representation that capture these differences for the problem of language identification. The current methods for language identification are falling in two main categories, textual information based and image-based methods. In the first case the assumption is that the textual information is available (digital born document, accurately OCRed documents or OCRed with consistent error) so the n-gram features will be used as representation for language identification. The second case methods which are only applicable on the document images composed of isolated letters are based on similar motivation, grouping of objects by defining some features, coding of the sequences and then matching it with textual data to find discriminant features. The first group of methods are very expensive to use and mostly not accurate and the second group is not applicable to cursive scripts such as Arabic. As mentioned before, patch based representation provide a good framework for representing elements of such scripts and NMTF, generalization of NMF to tri-factorization, provide a good base for representation, description and classification of Arabic script languages. Therefore, the third objective of this research is to propose a new method for language identification of Arabic script languages which will be detailed in Chapter 5.

2.2 General approach

New methods for document image categorization is proposed in this thesis, for a better understanding of relevant features. These new methods have direct links and together provide a framework for document image categorization at different levels of the layout and with different targets. Two main themes of this framework are representation and feature learning and for each problem this framework is adapted to that specific problem. The framework for document image categorization and its adaptation to different problems is illustrated in Figure 2.1. The main assumption of this framework is that the components of a document image are informative and information will be captured for the different task of categorization in different ways. As mentioned before, extracting these components is a difficult task so the first step of the framework to overcome this problem by patch representation. Patch extraction from a document image needs few parameters such as location, size and shape and these parameters can be adjusted and adapted to different categorization system. Feature extractions by human expert have many limitations and needs verification and/or modification if the target dataset changes. Therefore, the next step of the framework is feature learning so the output of patch representation will be explored and exploited to capture the variation in data that will be reflected in bases matrix. This step depends on the categorization problem and different strategies are needed. In script identification the differences in the shapes of the components of different scripts will be captured and used for categorization. In font and style identification the differences between the components of different categories are appeared details. Although, the discriminant information for script and style identification lie in the components, discriminant information for language identification is in how these components are used. Finally, the categorization step of the framework depends on the feature and the nature of features determine the categorization strategy.

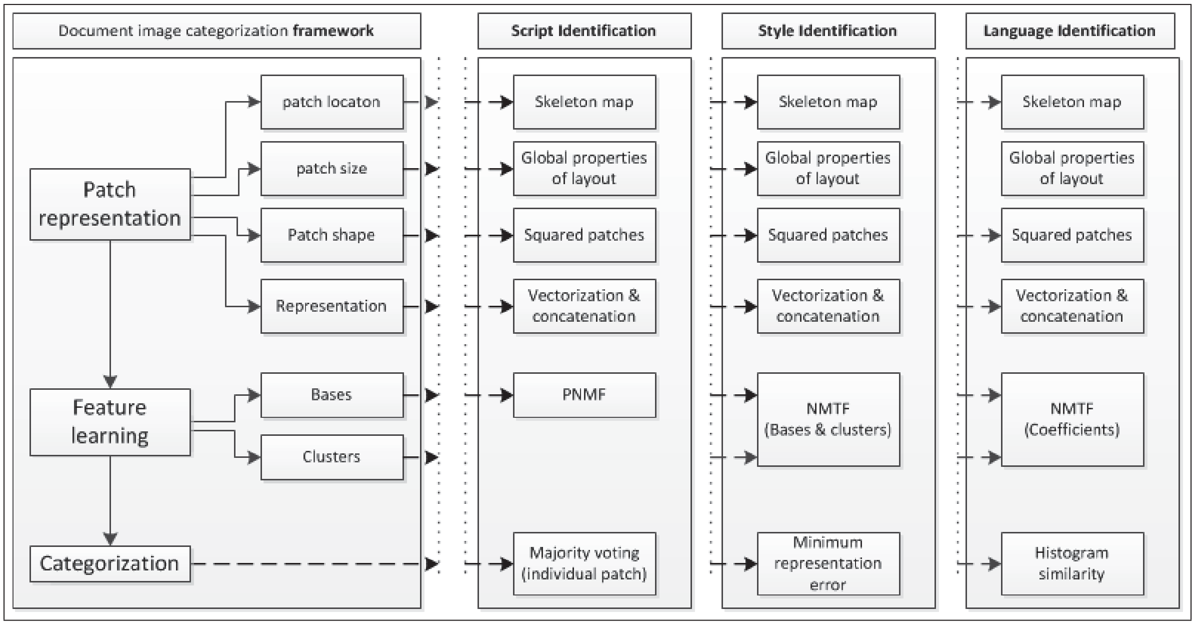


Figure 2.1 proposed framework and its application in different problems

2.2.1 Script identification

A new method has been proposed for script identification of ancient manuscripts. This method integrates the patch representation and NMF for representation and feature extraction. As mentioned, the patch representation is appropriate for representation of components of the script. Therefore, in patch representation step, some keypoints are extracted from the document image and then by using some estimation for patch size many patches are extracted from the document image. The keypoints locations are set based on skeleton map and patch size is set based some global property of document image, such as average text height, average stroke-width and average distance between text lines. The set of patches from different scripts are combined as data matrix then this set of patches are processed by PNMF algorithm. PNMF method which is proposed for low-rank approximation, factorize data into product of two matrices, bases matrix and coefficient matrix. The bases are learned from data in a collaborative way. This will help in dimensionality reduction, noise removal, abstract representation for the data. In most of method based NMF,the coefficient matrix is obtained in an iterative process but in PNMF the representation can be obtained by matrix production. Different methods for keypoint extractions are tested and it is observed that the keypoints extraction based on skeleton map has higher performance. The proposed method is compared with two script identification methods, one based on global approach and one based on local approach, and it outperformed both of them by good margins (Chapter 3).

2.2.2 Font and style identification

The second objective is realized by proposing a new method for font and style identification of printed and ancient manuscripts. In the proposed method, the patch based representation is used to convert image into many overlapped patches that are extracted using skeleton map of the document image. After binarization and skew estimation and correction, the skeleton map is obtained and keypoints (patch centers) are extracted from the skeleton map. In order to reduce the number of patches, priority for being keypoints is assigned to pixels based are on branch points, then the neighbor pixels are skipped. Patch size estimated with global measures of text height and stroke width for machine-printed manuscript and for ancient manuscript the iterative procedure is used. As ancient manuscripts have irregularities in layout, we set the parameter of patch size manually and extracted patches, then we use NMTF to learn bases and clusters. For the test set we estimated the parameter and by setting few values above and few values below the estimation we used NMTF for representation and the representation error determine which set of bases and cluster should be used as final representation. For font and style identification, we use NMTF for representation of each font and these bases are used to represent the patches of new image. The label of the set of bases and clusters that produces lowest representation error will be selected as label for test image. The proposed method is tested in different situations and the results are also compared to other method. The evaluation shows the performance of the proposed method and generalization power from machine-printed to handwritings (Chapter 4).

2.2.3 Language identification

The third objective proposes a method for language identification Arabic script languages. In this method document images with different languages are represented with the patches extracted by keypoints from the skeleton map and the size is set by global measure of document image like text height and distance between text lines. In order to create codebooks for representation, the clustering algorithm is needed. Due to high dimensionality of data, affection by noise and degradation and also presence of different fonts or styles, we used NMTF. The NMTF algorithm belongs the family of bi-clustering method that perform the clustering of samples and features simultaneously. These methods show improvement in clustering results compared to simple clustering methods such as Kmeans. One interpretation for NMTF algorithm is that it projects data into low-dimensional space and then cluster data and by alternating between these steps it refines the representation. In our work in which the data are patches, NMTF creates an abstract representation for patches and clusters the similar shapes (even different fonts will be clustered together). In order to consider variation of fonts, we samples patches of document images with different fonts and then used NMTF to create dictionary or codebooks. Then this dictionary will be used to represent patches of the image. The coefficient matrix obtained from this process has index like values (similar to Kmeans) due to orthogonality constraint in objective function of NMTF. Then the representation is obtained by summing the elements of this matrix along variables. This representation would be used to classify document images based on languages.

CHAPTER 3

ARTICLE I - PSI: PATCH-BASED SCRIPT IDENTIFICATION USING NON-NEGATIVE MATRIX FACTORIZATION

Ehsan Arabnejad[1], Reza Farrahi-Moghaddam[1], Mohamed Cheriet[1]

[1] Synchromedia Laboratory, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

Published in Elsevier Pattern Recognition
Volume 67, July 2017, Pages 328-339

Abstract

Script identification is an important step in automatic understanding of ancient manuscripts because there is no universal script-independent understanding tool available. Unlike the machine-printed and modern documents, ancient manuscripts are highly unconstrained in structure and layout and suffer from various types of degradation and noise. These challenges make automatic script identification of ancient manuscripts a difficult task. In this paper, a novel method for script identification of ancient manuscripts is proposed which uses a representation of images by a set of overlapping patches, and considers the patches as the lowest unit of representation (objects). Non-Negative Matrix Factorization (NMF), motivated by the structure of the patches and the non-negative nature of images, is used as feature extraction method to create low-dimensional representation for the patches and also to learn a dictionary. This dictionary will be used to project all of the patches to a low-dimensional space. A second dictionary is learned using the K-means algorithm for the purpose of speeding up the algorithm. These two dictionaries are used for classification of new data. The proposed method is robust with respect to degradation and needs less normalization. The performance and reliability of the proposed method have been evaluated against state-of-the-art methods on an ancient manuscripts dataset with promising results.

Keywords

Script Identification, Non-Negative Matrix Factorization, Patch Representation, Clustering

3.1 Introduction

A writing system or script is a tool to express concepts in any languages by using graphical shapes and symbols to represent elements. Each script or writing system has a set of defined elements called characters (alphabets) and each element is assigned a special meaning and rule in interpreting the language. Although; some writing systems have same origin, over time evolution and changes makes them different. The writing systems of the world can be classified into a few groups; the generally accepted categorization (Daniels & Bright, 1996) is shown in Figure 3.1.

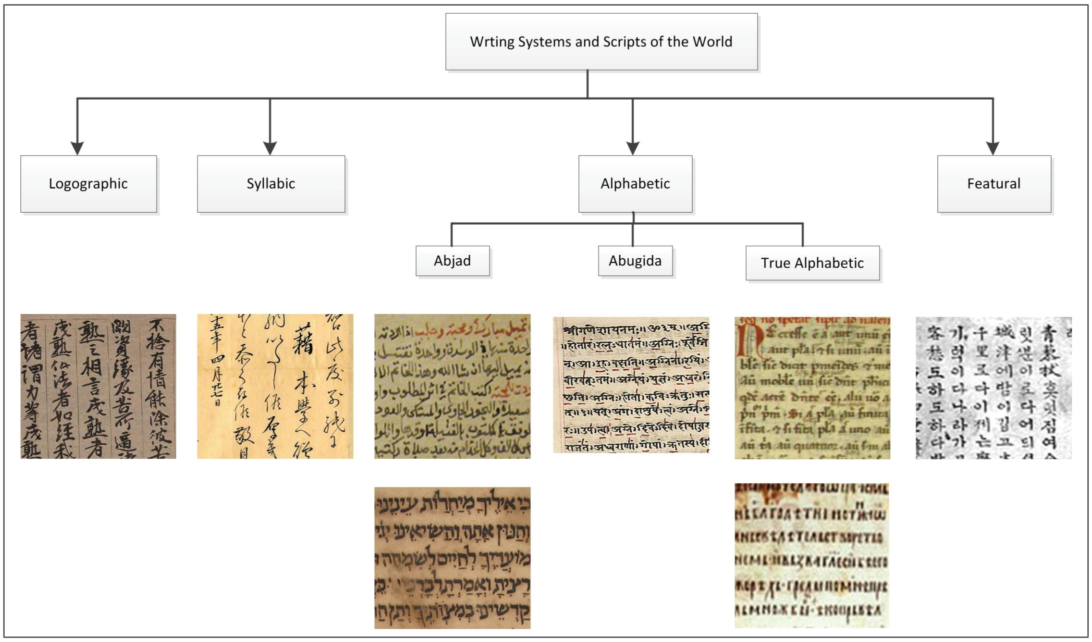


Figure 3.1 Writing systems and scripts of world

In recent years several word/character recognition systems have been proposed and developed to handle the demands of the digital world. But all of these systems are limited to working with specific scripts or styles. This means that they can produce promising results if they are used to recognize the components of a particular class of document image. Different scripts, however, use different graphical shapes, symbols and writing rules (script grammar) to construct statements. Hence, using only one word/character recognition system for processing different types of document images is not efficient.

An effective strategy to deal with this problem would be to use a bank of word/character recognition systems- where each recognition system is adapted to specific a script- combined with a script identification system. The script identification system selects a proper word/character recognition system based on the script of the document. This system is also needed when the document image is composed of different scripts. The first step of script identification is to extract some features from a set of document images; the second step is to use those features for classification.

In the past, several different styles were invented and used for writing of a specific kind of script with the aim of fast writing, artistic purposes and/or reducing ambiguity. Each style has its own specification, such as the shapes of the letters in a word, the types of connections between letters, variations in the shapes of one or more characters, overlap between words, etc. In addition to this challenge, there are other challenges that make a script identification of an ancient manuscript a difficult task, such as unconstrained layout, degradation due to aging and bad maintenance, noise and additional degradation imposed on document images in the digitization process.

Script identification methods can be divided into two categories based on the type of features:

- Structure-based methods make the assumption that structure, connections and the writing style of different scripts are good features to differentiate between them. In most of these methods, connected components (continuous blobs of pixels) are analyzed according to their shapes, structures and morphological characteristics to produce discriminant features.

- The visual appearance-based methods make the assumption that differences in the shape of individual characters, grouping of characters into words and grouping of words into sentences can be tracked without actually analyzing the pattern of the characters in the document. Several methods have been proposed to capture these properties and produce good features for script identification.

On the bases of the level of applying a method inside documents, the script identification methods can be classified into 4 groups:

- Page-wise methods such as the method in (Dhandra *et al.*, 2006)
- Paragraph-wise (or block-wise) methods such as the method in (Busch *et al.*, 2005)
- Text-line-wise methods such as the method in (Aithal *et al.*, 2010)
- Word-wise methods such as the method in (Roy *et al.*, 2010)

The limitation of the structure-based method is that the extracted features are sensitive to noise and degradation, whereas the limitation of the visual-appearance based method is it a great deal of normalization. The common limitation of both categories is that all of the methods are proposed for a specific level of layout and cannot be used for different layout levels. In this work, we propose a new method for script identification of ancient manuscripts based on patch representation and Non-Negative Matrix Factorization(NMF). This method is robust to degradation and noise and is flexible inasmuch as it can be used with different levels of layout for script identification.

The organization of this paper is as follows. In section 2, related state-of-the-art methods are discussed. Then, the proposed method is presented in section 3. The experiments and evaluation are presented in section 4. Finally, the summary of the work is presented in the conclusion.

3.2 Related Work

Here, we briefly introduce some leading and pioneering methods for script identification of document images. The structure-based method is mostly applied on connected components, and these objects are analyzed to extract discriminant features for script identification: In Pal *et al.* (2003) the hierarchical script identification method for Indian documents was proposed. The features were headlines, horizontal projection profile, water reservoir principle based features, left and right profiles and features based on jump discontinuity. In Hangarge & Dhandra (2008), two sets of global and local features and a decision tree strategy are used for script identification. The global features are stroke density and pixel density; and the local features are aspect ratio, eccentricity, extent and directional profile densities. In Chanda *et al.* (2009), a set of 8 features that describe the properties of connected components combined with the Kernel-Support Vector Machine (SVM) are used for word-wise script identification. In Rezaee *et al.* (2009), features of distribution of horizontal projection which is different for Latin and Persian are used. In Aithal *et al.* (2010), features of first and second maxima of horizontal projection combined with a rule-based classifier is used for script identification. In Chanda *et al.* (2010), chain-code histogram-based features inspired by the observed visual differences between Han-based scripts such as Chinese, Japanese and Korean are proposed. These scripts have dissimilarities which can only be found in small parts of the characters and chain code is used to capture this difference. In Gopakumar *et al.* (2010), 9 features from text line and 9 features from 3 horizontal zones of text line as well as the Euler number of text lines are used for script identification. In those works the feature selection is used to increase the classification rate. In Roy *et al.* (2010), a set of features from connected components is obtained by concatenation certain features, such as the average aspect ratio or distributions of pixels around the center of mass. Then K-NN is used to classify new features. In Khoddami & Behrad (2010), a script identification method for separation of Persian and Latin script based on curvature scale space is proposed. In Marinai *et al.* (2010), script identification method based on bag of characters and self-organizing map (SOM) is proposed. The weighted frequency of symbols (as features) and Self Organizing Map (SOM) with the distribution of pixels as input is used for clustering symbols. The visual-appearance based method is mostly inspired by texture classification/segmentation methods and uses the texture features as discriminant information for script identification:

In Busch *et al.* (2005), the authors propose a method for block-wise script identification using various texture features. These features are the Gray-Level Co-occurrence Matrix (GLCM), Gabor energy features, wavelet energy features, wavelet log mean deviation features, wavelet co-occurrence signatures and wavelet scale co-occurrence signatures; and they are used to separate Latin, Chinese and 6 other scripts. In Pan *et al.* (2005), rotation-invariant features for block-wise script identification are proposed. These features are obtained after using steerable Gabor filters and 1-D Fourier transform. Their classifier is a two-layer feed-forward neural network with rejection strategy. In Ben Moussa *et al.* (2008), a script identification method based on fractal features are proposed for printed and handwritten documents written in Arabic and Latin scripts. The researchers use K-NN and Radial basis function to separate different classes. In Chanda *et al.* (2010), the authors propose a word-wise script identification that works in two steps. The first step is to extract 64-dimensional chain-code histograms of a bounding box and the second step is to extract 400-dimensional feature vector from the magnitude of gradient with a Sobel filter. The Kernel-SVM classifier is used for classification. In Hiremath *et al.* (2010a) and Zhou *et al.* (2010) texture-based features are used for script identification. In those methods, wavelet transform and features based on co-occurrence histogram and wavelet energy histogram are used for the classification of different scripts. In Pan & Tang (2011), a script identification method based on global and local texture features is proposed. In the first step, the image is decomposed into different intrinsic mode functions (IMF) by a method called Bi-Dimensional Empirical Mode Decomposition (BEMD). In the second step, Local Binary Pattern (LBP) is used to detect the local features of the textures. In the last step, SVM is used to classify the features of the new image. The drawback of structure-based features are that extracting accurate features requires prior information about the type of script and these features are sensitive to noise. Texture-based approaches need intensive pre-processing and normalization to prepare the document image for feature extraction, and some of steps are also script dependent. Most of the proposed methods have been tested on machine printed documents where the contrast between text and background is high, interference by noise and degradation are low and the structure of the text has been well preserved in the digitization process. The problem of script identification is more challenging when the target is ancient manuscripts where the challenges are various writing styles and variations in the manuscript, a high level of noise and the degradation and unconstrained structure of text.

3.3 Motivation

In this paper, we address the problem of script identification in ancient manuscripts by proposing a novel, robust and flexible method. To our knowledge no method has been proposed for these kinds of documents in which many challenges should be faced and resolved to produce accurate and acceptable results. Some of these challenges are different writing styles of some scripts, unconstrained layout of document image, handwriting variations, presence of noise and imposed degradation. Here we propose a method for script identification that is based on Non-Negative Matrix Factorization (NMF), with the intuition that NMF provides a robust framework for representation and feature extraction.

Despite the fact that millions of words are available in each script, they are composed of a set of limited elements (letters) and in many situations these elements are discriminant features for script identification. To extract these components as features, many steps, such as layout analysis and segmentation are needed. This is an active field of research and many people are working on it, but the assumption is that the script of the document is known.

Here we propose to use image patches to avoid the error of segmentation and also to overcome the drawbacks of the previous methods. The first advantage is that patches are less sensitive to noise or discontinuities in connected components (unlike the structure-based approaches that use connected components). The other advantage is that they do not need intensive normalization (unlike the texture-based approaches). The third advantage is that segmentation is not needed; and if the patches and their size are chosen properly, they will be good approximations of elements of a particular script or of a combination of elements. As a result, the script of a document image can be recognized by proper comparison of its patches to the set of labeled patches.

Template matching is the direct way to compare patches and classify them; but various challenges, such as writing styles, handwriting variations, degradation and noise, reduce the performance of the template matching approach. To overcome the drawback of template matching, we propose to use Non-Negative Matrix Factorization (NMF) as a representation and then to use this representation for classification.

NMF is used in many applications of text processing and mining of text collections using textual data. The main difference between the work proposed here and these methods is the type of data. Here we propose a method for script identification of document images in image collections using patches and pixel values.

3.4 Notation

The following notation is used in this paper:

EUD	Euclidean Distance
KLD	Kullback-Leibler Divergence
F	Ferobenius Norm
X	Matrix of Feature vectors with the size of $m*n$
x	feature vector with the size of $m*1$
W	Weight Matrix with the size of $k*n$
H	Bases Matrix with the size of $m*k$
P	Projection Matrix with the size of $m*m$
U	Bases Matrix with the size of $m*k$

m	dimensionality
n	number of samples
k	number of bases
l	number of cluster centers
c	Cluster center
i	index
j	index
t	Iteration index

Image patch P is defined as a window centered at p with this property:

$$P_{p,w,q} = \{z | z \varepsilon \Omega, \|x - z\|_q \leq w\} \tag{3.1}$$

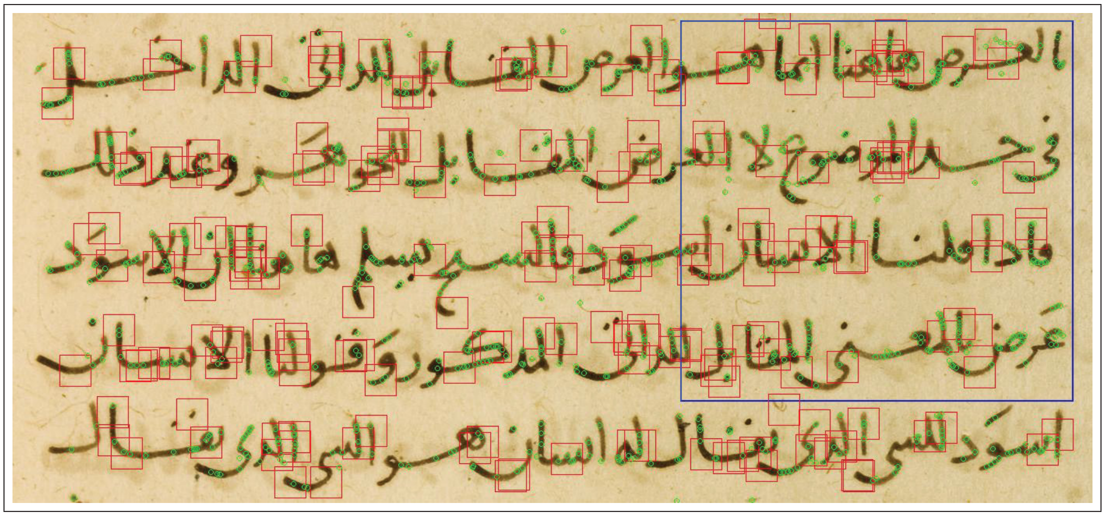
where Ω is the neighborhood of pixel p, z are pixels that are in the neighborhood of p, w is the size of the window and $\|.\|_q$ is L_q norm that defines the shape of the window. In our case patches are rectangular windows centered at pixel p with the size of $((2*s)+1)*((2*s)+1)$ pixels.

In Figure 3.2 the different levels of information that have been used and discussed in this paper are shown. The figure shows a part of an image, the blue rectangle is an image block, the red rectangles are sample patches, and the green circles are the center of patches; for better visibility only a subset of patches is shown.

3.5 Non-Negative Matrix Factorization (NMF)

3.5.1 Background

One common ground in the various approaches for noise removal and model reduction is to replace the original data by a lower-dimensional representation using subspace approximation. Often the data to be analyzed is non-negative, and this property should also be imposed on low



العرض هاهنا انما هو العرض المقابل للذاتي الداخل
في حد الموضوع لا العرض المقابل للجوهر وعند ذلك
فاذا قلنا الانسان اسود فالسود يسمى هاهنا الاسود
عرض المعنى المقابل للذاتي المذكور وقولنا الانسان
اسود للشي الذي يقال له انسان هو الشي الذي يقال

Figure 3.2 Levels of information: part of image, image block (blue rectangle), image patches (red rectangles), center of the patches (green circles)

rank approximation. This is the so-called Non-Negative Matrix Factorization (NMF) problem and can be stated in generic form as follows: Given a non-negative matrix (data matrix) $X \in R^{m\times n}$ and a positive integer $k < min(m,n)$, the goal is to find non-negative matrices $H \in R^{m\times k}$ and $W \in R^{k\times n}$ such that:

$$X = HW \tag{3.2}$$

Generally, the product HW is not necessarily equal to X and is merely an approximate factorization of a rank at most k, and a similarity measure or an objective function is needed to quantify the similarity between original data and its approximation.

Common similarity measures are Euclidean distance (EUD) Lee & Seung (1999) and Kullback-Leibler divergence (KLD) Kullback & Leibler (1951) used in Lee & Seung (1999). Since, the objective function is data dependent, other similarity measures are used such as, the Minkovsky family of metrics or l_p-norm, earth mover's distance Sandler & Lindenbaum (2011), $\alpha-$ divergence Cichocki *et al.* (2008), $\beta-$ divergence Kompass (2007), $\gamma-$ divergence Cichocki *et al.* (2006), Bergman distance Dhillon & Sra (2005), and $\alpha - \beta -$ divergence Cichocki *et al.* (2011). The robustness of some similarity measures has been shown on some datasets.

The converge to unique solution or stationary local minimum is not guaranteed in the basic NMF algorithm Lee & Seung (1999), so additional constraints for the coefficients and/or bases matrices are used to solve this issue, such as sparsity constrains in Hoyer (2002), orthogonality constraints of bases in Choi (2008), discriminant constraints in Wang & Jia (2004), and manifold regularity constraints in Cai *et al.* (2011). In structured NMF, the factorization (not the constraints) is modified to produce the desired result, for example by weighed NMF in Kim & Choi (2009), convolutive NMF in Smaragdis (2007) and Non-Negative Matrix Tri-Factorization in Yoo & Choi (2010).

To use the good properties of NMF in a broad way without any constraints on the data, generalized NMF algorithms have been proposed, such as Semi NMF in Ding *et al.* (2010), Non-Negative Tensor Factorization in Shashua & Hazan (2005), Non-Negative Matrix-set Factorization in Li & Zhang (2007) and Kernel-NMF in Lin (2007).

In most cases, the choice of the objective function and algorithm is highly data dependent; the nature of data determines the proper algorithm and objective function.

The original NMF is solved by multiplicative algorithm in Lee & Seung (2001), where two multiplicative update rules are used to minimize the objective function iteratively and alternatively:

$$W_{kj}^{(t+1)} = W_{kj}^{(t)} \frac{(H^T X)_{kj}}{(H^T H W^{(t)})_{kj}} \tag{3.3}$$

$$H_{ik}^{(t+1)} = H_{ik}^{(t)} \frac{(X W^T)_{ik}}{(H^{(t)} W^T W)_{ik}} \tag{3.4}$$

where t is the iteration index.

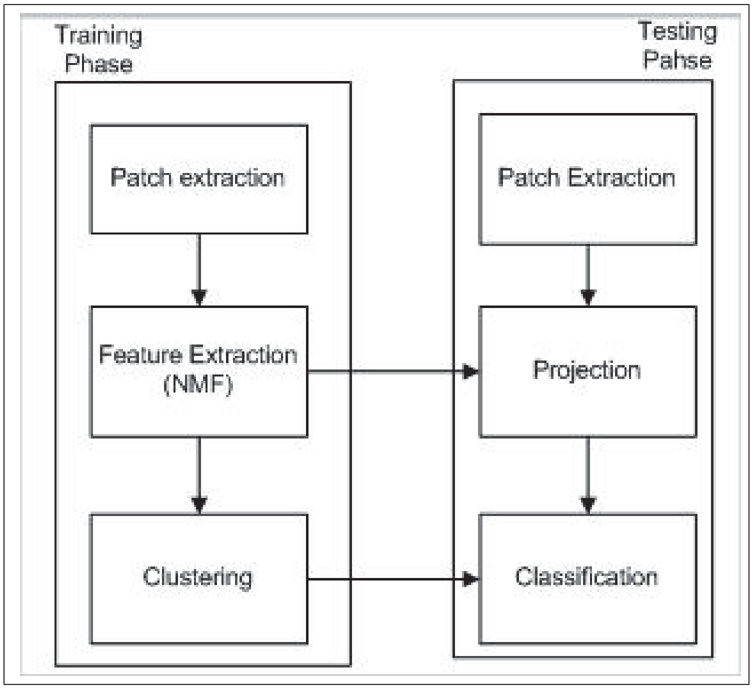


Figure 3.3 Flowchart of the proposed method

3.6 Patch-based Script Identification Using NMF

The proposed method for script identification is illustrated in Figure 3.3. After pre-processing and preparing the data, several patches are extracted from the image blocks, and then NMF is used to learn and extract some bases from this set of patches. For the test set, a similar process is used for extracting the patches, and the bases that are learned in the training step are used to extract features. Then, the K-nearest neighbour algorithm (K-NN) is used to classify new features on the bases of proximity to the nearest feature point in the training set. After assigning a label to each patch, the label of the image block is estimated by majority voting using the labels of all of the extracted patches from that image block.

3.6.1 Pre-processing

To prepare data for script identification, we manually extract some image blocks from different document images. These image blocks contain 4 text lines that cover all of the width of the image blocks. These image blocks are converted to gray level images and then converted to binary images (black and white) using Otsu's Otsu (1979) method. These image blocks are skew corrected by analyzing the variance of projection of the pixels from different angles. This is based on the fact that when the text lines are completely horizontal, the difference between maximum value in the projection (showing the position of the baselines) and the minimum value (showing the locations between text lines) is high.

Although our work does not need any other pre-processing step, in order to be able to compare our work fairly with state-of-the-art methods, we use other pre-processing steps to create a normalized dataset. The first step is normalization of the image blocks to fixed size, then normalization of text height as well as distance between baselines and, finally, stroke-width normalization. These normalization steps are needed to prepare the data and apply the method based on texture (Busch *et al.*, 2005). This normalized dataset is only used for comparing 3 script identification methods.

3.6.2 Patch extraction

We have tested different methods to extract patches from the image blocks, such as random sampling, uniform sampling, and with/without overlap to find the best method for document images. Also, methods such as Scale Invariant Feature Transform (SIFT) Lowe (1999) or Speeded Up Robust Features (SURF) Bay *et al.* (2008), which have been proposed to detect corner points and create descriptors for those corner points, were tested for keypoint extraction. Because for document images the position of the text is approximately known, here we use the skeleton map of connected components Farrahi Moghaddam *et al.* (2012) with each pixel on the skeleton map as a center point for the patches. This approach reduces the number of extracted patches in contrast to the situation which all the foreground pixels (text) are used as keypoints. Ancient documents are degraded by lots of noise, which creates the unwanted patches. Some scripts also have dots or diacritics that are not valuable information, so these pixels and components have to be removed before patch extraction. Individual pixels and small components are removed by analyzing the area of those components and using a threshold to eliminate unimportant pixels. The patch size is estimated by three measures: text height, distance between baselines and stroke width. These parameters are estimated automatically and easily from the binary image. In the SIFT and SURF methods the goal is to find the corner points; however, there exist many points that have discriminative information, but these points are skipped by the SIFT/SURF methods because they are not corner points. In Figure 3.4, sample image blocks and detected keypoints using skeleton, SIFT and SURF keypoint detectors are shown. These 3 methods are applied on the same image with the same level of pre-processing. After patch extraction, these patches are resized to the same size and converted to vector to create Data matrix X. Considering the resolution of images and number of text lines, we set this value as 33, which results in a dimensionality of around 1000.

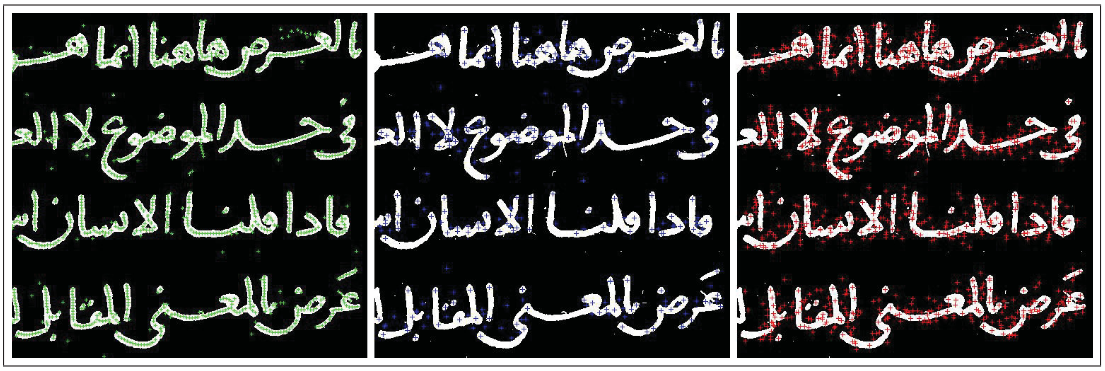
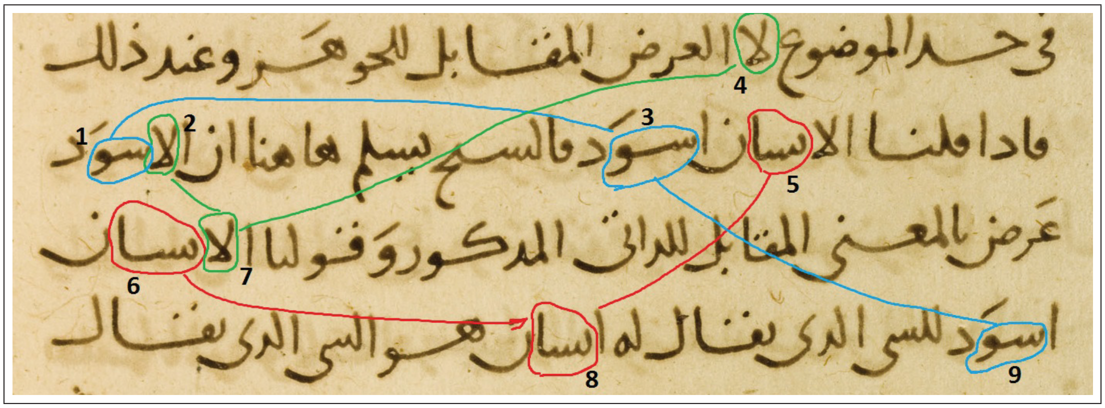


Figure 3.4 Keypoints detected by Skeleton method (left), SIFT method (middle) and SURF method (right)

3.6.3 Feature extraction - dimensionality reduction

Patch-based algorithms are defined by the processing data in finite dimensional windows that contain a subset of the pixels within an image, and this subset is considered separately from the rest of the image. For document images these subsets of pixels could be part of a character, a character, a combination of several characters or part of a word, depending on the size and position of the patch. As scripts and writing systems are composed of a limited set of letters or symbols, the similarity of different patches is high if they contain the same or similar information about character or characters. Hence, if some parts of a text have been distorted by degradation, other similar or highly similar patches can be used to retrieve correct information.

We can see this similarity in Figure 3.5 where connected components determined by similar color are the same (also connected by arrows). For the machine printed-document there is no variation other than noise or possible degradation but as we can see from the figure, for this sample manuscript, the components of 1-3-9 , 2-4-7, 5-6-8 are the same; but because it is handwriting, there are lots of variations. For machine-printed documents the structure of the text is well defined and this correlation can be tracked by straightforward template matching. For ancient manuscripts, however, as mentioned before, various conditions limit the usefulness of a simple and direct template matching method. In order to overcome this drawback, a more complex and advanced approach is needed to deal with the ancient manuscripts. From another point of view these patches are two-dimensional areas of original images; and even if the size of the patches is set to a few pixels, the resulting feature vector will be high-dimensional, so processing of these patches with a direct method is computationally expensive. In order to overcome all of these drawbacks, we propose the use of NMF.



في حد الموضوع لا العرض المقابل للجوهر وعند ذلك
فاذا قلنا الانسان اسود فالسطح بسيط هاهنا ان الاسود
عرض بالمعنى المقابل للذاتي المذكور وقولنا الانسان
اسود للشي الذي يقال له انسان هو الشي الذي يقال

Figure 3.5 Occurrence of similar connected components in a sample document image for handwriting

One of the main properties of NMF is part-based reconstruction,which is obtained by a non-negativity constraint on the objective function. This property is well suited for representation of an object in an additive manner and depicts a real physical object. In our case, where the lowest level of information is patches and these patches contain parts of characters/words, this representation is a natural choice. Another property of NMF is collaborative learning, which means that several parts contribute to learning the bases. These properties provide an adequate intuition to use NMF for representation and feature extraction of ancient manuscripts that are affected by degradation and noise. If any part of a patch is thus corrupted, the collaborative learning will retrieve the correct information from similar patches and reduce errors in the representation of patches.

The sparsity property of NMF allows a subset of patches to contribute to learning so each class of script and the corresponding component will contribute to learning some of the bases; and the representation obtained by NMF will be discriminative. From another point of view, our feature vectors have a dimensionality of around 1000 or more, which is high for processing and classification; so NMF, which is suggested for low-rank approximation, can be used to reduce the dimensions of the data. This increases the performance of classification, speeds up the algorithm and avoids the curse of dimensionality in the subsequent steps. Higher performance and robustness against noise, degradation and writing style are other advantages of the proposed method. To on our knowledge, there exists no method for script identification of ancient manuscripts based on NMF. Here we propose to use Projective Non-Negative Matrix Factorization (PNMF) Yuan & Oja (2005), which is based on the idea that the NMF approximation is a projection of the matrix of data X with projection matrix P.

$$X = PX \tag{3.5}$$

The projection matrix P can be defined as

$$P = UU^T \tag{3.6}$$

and each column in U can be interpreted as a base, so U is the matrix containing all of the bases and the EUD objective function for PNMF is:

$$\min_{U>=0} \frac{1}{2} \left\|X - UU^T X\right\|_2^F \tag{3.7}$$

The PNMF algorithm has some advantages compared to NMF, such as orthogonality of bases that help in better convergence, static local minimum (experiments are reproducible) and closer relation to clustering.

$$U_{ij}^{(t+1)} = U_{ij}^{(t)} * \frac{2\left(XX^T U^{(t)}\right)_{ij}}{\left(U^{(t)} U^{T(t)} XX^T U^{(t)}\right)_{ij} + \left(XX^T U^{(t)} U^{T(t)} U^{(t)}\right)_{ij}} \tag{3.8}$$

The normalization of matrix U shown below is used to make the PNMF algorithm stable and avoid oscillation.

$$U_{new} = \frac{U}{\|U\|_2} \tag{3.9}$$

Here is the objective function and multiplicative update rule for KLD :

$$D(A||B) = \sum_{i,j} (A_{i,j} log \frac{A_{i,j}}{B_{i,j}} - A_{i,j} + B_{i,j}) \tag{3.10}$$

$$U_{ij} = U_{ij} * \frac{\sum_k ((U^T X)_{jk} + \sum_l U_{lj} X_{ik})}{\sum_k X_{ik} ((U^T X)_{jk} / (UU^T X)_{ik}) + \sum_l U_{lj} X_{lk} / (UU^T X)_{lk}} \tag{3.11}$$

In Figure 3.6 one sample patch and its reconstructions with a different number of dictionaries are shown. The image on the left is the original patch and the rest are reconstructions with 10, 40 and 150 bases respectively. We can see that allowing more bases to contribute to reconstruction of patches creates a better representation.

In Figure 3.7 and Figure 3.8 the average representation error by using a different number of bases is shown for PNMF with Euclidean distance objective function EUD and Kullback-Leibler divergence objective function (KLD). The comparison between the representation errors of the training set and the test set in Figure 3.7 shows that the representation errors of two sets are very close and the bases are highly informative.

In Figure 3.9 the comparison between two objective functions is shown. The representation errors of two objective functions are not directly comparable, so the error measure with EUD is used. We can see that the EUD has a lower representation which is obvious because of the relation between the objective function and the error measure. In order to see the difference between two objective functions at the patch level, some sample patches and the reconstruction with a different number of bases are shown in Figure 3.10. KLD performs better when a lower number of bases are used; but by using a higher number of bases EUD shows less representation error.

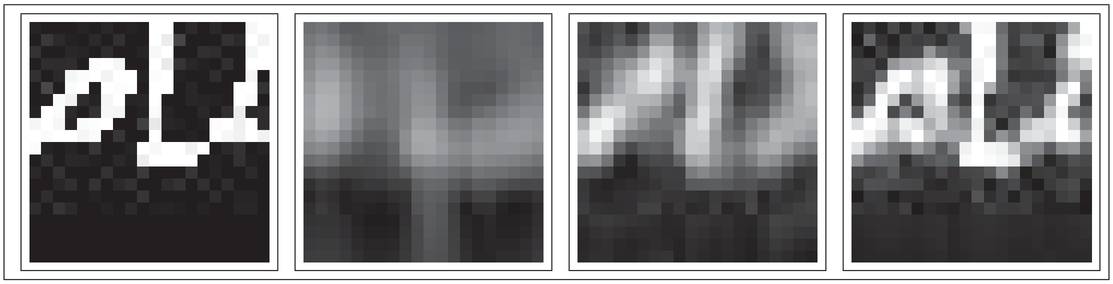


Figure 3.6 Original image and its reconstructions using 10, 40 and 150 bases obtained by NMF

From the feature extraction point of view, NMF (generally) extracts the common information that are provided by patches and represents it in the bases matrix. This information depends on the component of a particular script; and since the components are different in various scripts, the provided bases and the representations are discriminative.

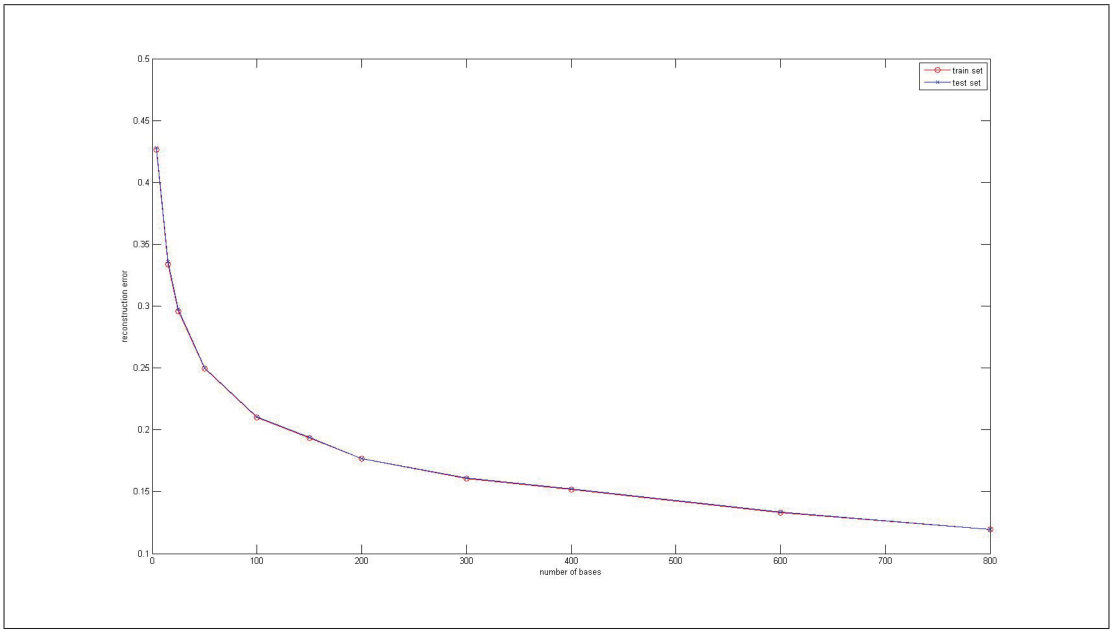


Figure 3.7 Average representation error for the train set and the test set with different number of bases (EUD objective function)

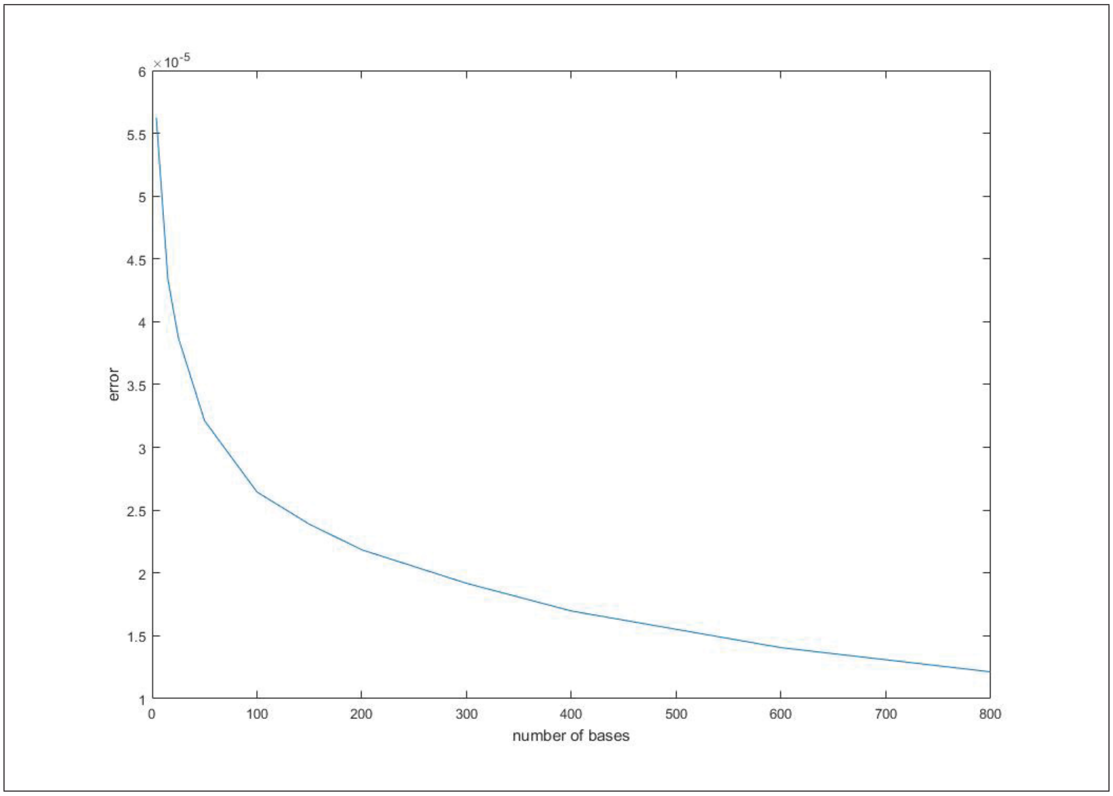


Figure 3.8 Average representation error for train set with different number of bases for KLD objective function

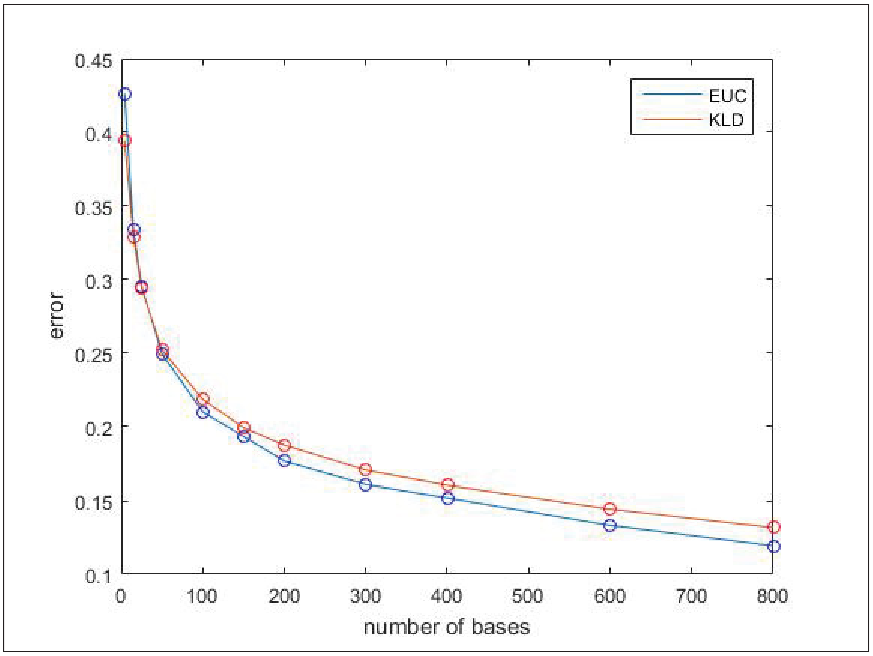


Figure 3.9 Comparison of representation error for EUD and KLD objective function measured by Euclidean distance

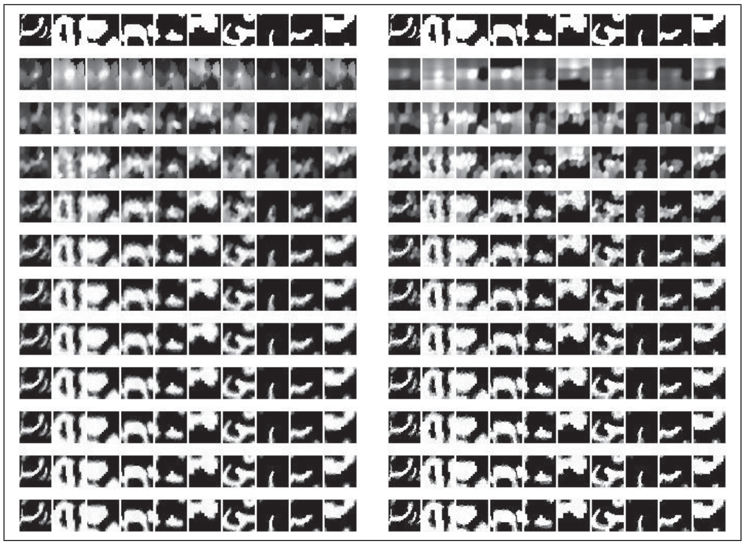


Figure 3.10 Comparison of representation error for two objective function in patch level, left) EUD objective function, right) KLD objective function

3.6.4 Clustering

Considering patches of 800 images blocks results in a data matrix of nearly 250,000 samples (75,000 as training and 170,000 as test). Even with a dimension of 50 the classification and calculation of Euclidean distance needs a great deal of memory and the algorithm will be very slow. Using the fact that for each script, words are composed of a limited set of characters and the fact that every patch contains a character or part of a character, we can use a clustering algorithm to reduce the number of patches to a reasonable number. This will increase the speed of the algorithm and, if it is used properly, also improve the performance of the algorithm. Here, the K-means algorithm with the objective function shown below is used:

$$J = \sum_{j=1}^{l} \sum_{i=1}^{n} \|x_i^{(j)} - c_j\|^2 \tag{3.12}$$

where x is the feature vector, c is the center of the cluster and l is the number of the cluster center.

3.6.5 Classification

Each new patch (from the test set) is classified by finding the nearest patch from the training set, and the label of the nearest patch will be assigned to it. As an image block consists of many patches, the labels of these patches can be used to estimate the labels the for image blocks. These labels can then be combined in various ways. Here we use a simple majority vote to predict the labels of image blocks.

3.7 Experimental Results

The proposed script identification method is evaluated on our dataset. Different experiments are performed to show the performance and robustness of the proposed method.

3.7.1 Database

To evaluate the method, a multi-script database was created using document images from 4 different scripts Arabic, Cyrillic, Hebrew and Latin. Around 800 document images were selected for this experiment; 40 percent of them were used as the training set and the rest as the test set. These document images were obtained from different digital libraries and belong to different periods of history. Our aim is to develop a database with different scripts, different writing styles and different languages. Each document image in this database was labeled according to its script. The images in our dataset have different resolutions and the resolution changes from 3900 * 3060 pixels in higher-resolution images to 600 * 450 pixels in lower resolution images. Our dataset is composed of different images with different numbers of text lines (from 15 text lines to 35) and different resolutions (the heights of the images vary from 600 pixels to 3900 pixels).

3.7.2 Experimental Results

As discussed in previous section, selecting a good value for k is very difficult and almost completely data dependent. Here, we tested different values for the number of bases in NMF algorithm and different values for the number of cluster center l. In Tables 3.1 and 5.3 the performance and the time complexity of the algorithm (in second) for high-dimensional space and low-dimensional space are shown, respectively. As can be seen, the time required to classify around 170,000 patches was reduced from 23,839 seconds to around 20 seconds in low-dimensional space. The performance of individual patch-based classification decreased by 2 percent in low-dimensional space compared to high-dimensional space but the classification of image blocks improved. The reason is that there are many patches that affected by noise and degradation and these patches are outliers. NMF extracts common information from the data so the representation obtained for these patches is not accurate and decreases the overall performance of patch-based classification. At the same time, however, the correct classification of effective patches (the patches that contain important information) improves in low-dimensional space. Therefore, the performance of block-based classification increases in low-dimensional space. On the basis of the experiment and results shown in Table 5.3, the best classification performance is obtained by using 200 bases obtained by the EUD objective function. In Figure 3.11 the performance of the algorithm with a different number of bases and different number of clusters is shown. According to this figure, the more stable performance obtained by using 1000 clusters. Figure 3.12 shows the time complexity of the algorithm and how it changes when different numbers of bases and clusters are used.

Because of the similarity of the proposed method with the SURF and SIFT method, we compared these 3 methods from two aspects; SURF/SIFT as descriptor and SURF/SIFT as keypoint extractor. In Tables 3.3 and 3.4, the performances of script identification using SIFT and SURF methods are shown, respectively. It can be seen that the performances at the feature vector level and also at block levels are lower compared to the proposed method.

In Figure 3.13 a comparison of the performances of two different key point detectors (skeleton keypoint and SIFT keypoint detector) is shown (the SURF keypoint detector is omitted because of lower performance). The patches are extracted as before by using the 3 measures from image blocks. These experiments are done using a combination of different numbers of bases [from 4 to 800] and different numbers of cluster centers (from 50 to the number of data points). Experiments 1 to 9 correspond to the very low value of the number of bases (here 4). We can see that when the lower value is used for the number of bases (for patches representation by NMF), the SIFT patches produce better results; but when a larger number of bases are used the skeleton patches outperform the SIFT patches. It is hypothesized that as the corner points that are extracted by the keypoint detector in SIFT have more common information than the skeleton patches, they need a lower number of bases for representation but they are not as discriminative as the skeleton patches when a higher number of bases are used. We can see that the classification performance drops when the low number of clusters is used while the number of bases is fixed. For SIFT patches this drop is less than for the skeleton patches when few bases are used; but with a higher number of bases, the skeleton patches show better results.

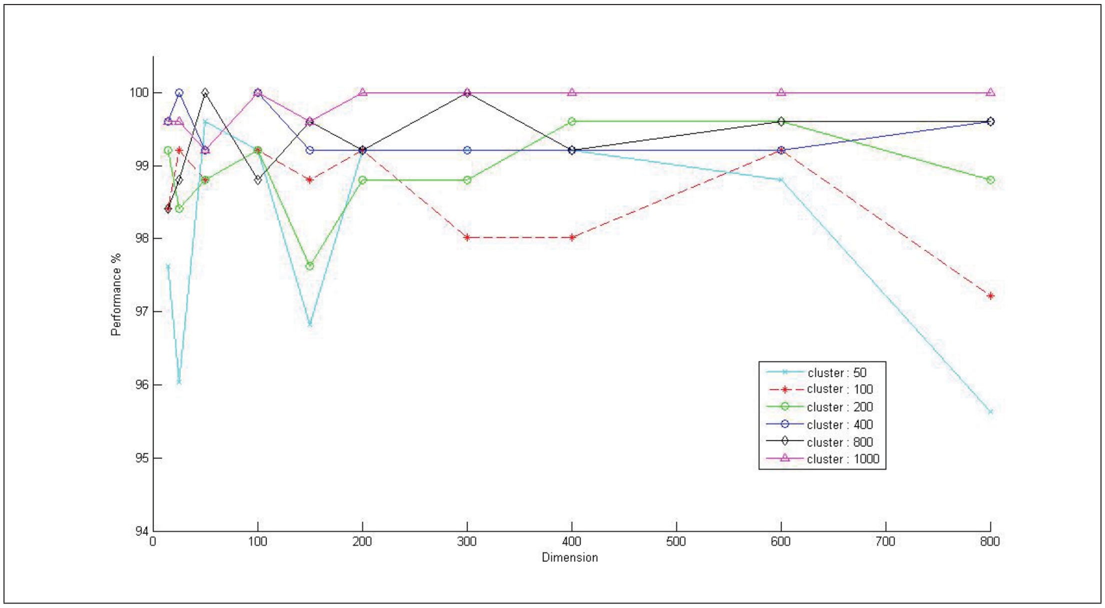


Figure 3.11 Result of classification with different numbers of bases (Dimension) and clusters

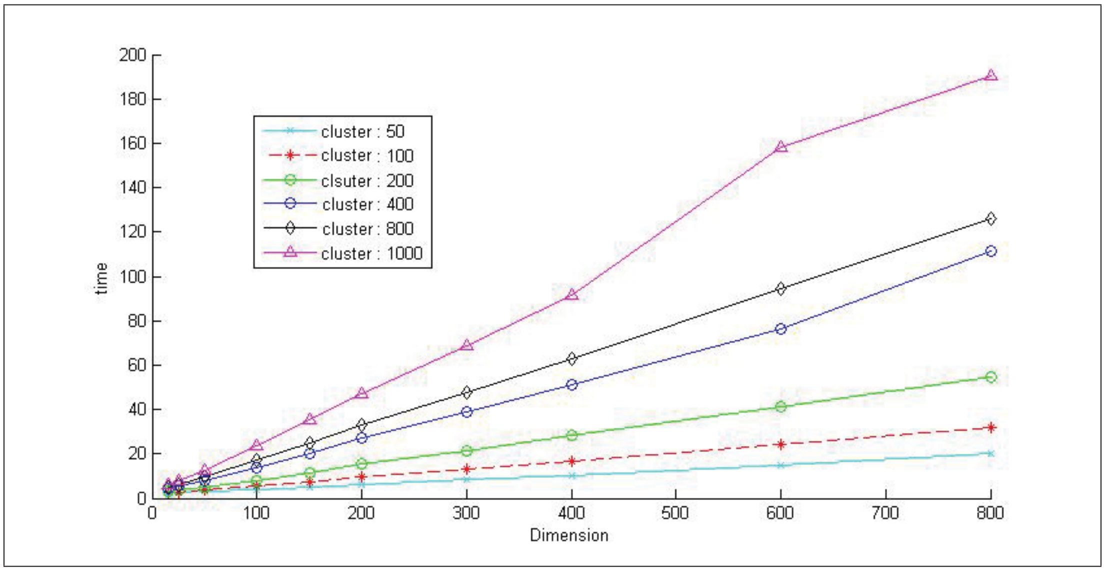


Figure 3.12 Time complexity of algorithm with different number of bases (Dimension) and clusters

In Figures 3.14 and 3.15 cluster centers that obtained by choosing 100 and 400 clusters are shown. When a small value is set for the number of cluster centers, the obtained cluster centers are not representative of the dataset with different scripts. However, when a proper value is set

Table 3.1 Performance of algorithm with different numbers of clusters in high-dimensional space (dimension of 1089) with 170,000 patches as test set, 75,000 patches used for cluster learning

No. of clusters	Performance(%) (Patch)	Performance(%) (Block)	time (s)
100	63.30	98.41	29.9
200	66.18	98.80	55.4
400	68.63	100.00	109.32
800	70.60	99.21	247.60
1 000	71.08	99.60	323.86
full dataset (75,000)	73.03	99.21	23839.32

Table 3.2 Performance of algorithm with different numbers of clusters in low-dimensional space (200 bases for PNMF) with 170,000 patches as test set, 75,000 samples for PNMF learning and cluster learning

No. of cluster	Performance(%) (Patch)	Performance(%) (Block)	time (s)
50	57.40	97.62	5.93
100	61.66	99.60	6.77
200	64.14	98.81	10.38
400	66.27	98.81	10.75
800	68.01	99.21	16.53
1000	68.72	100.00	19.19
full dataset (75,000)	74.23	100.00	1448.03

for cluster learning, the obtained cluster centers and the coverage of different scripts and styles do better and the resulting dictionary improves classification performance.

To show the performance of the proposed method against other methods, the method based on texture feature (Busch *et al.*, 2005) and another method based on shape codebook Zhu *et al.* (2009) were implemented and used for script identification. For a fair comparison, the normalized dataset is used here. In Table 3.5 the classification performance and comparison between algorithms are shown, and it can be seen that the proposed method performs better in the dataset of ancient manuscripts and that results validate the performance of the proposed method.

Table 3.3 Performance of algorithm with different numbers of clusters using SURF features (94,000 samples as test , 62,000 samples for cluster learning

No. of cluster	Performance(%) (feature)	Performance(%) (Block)
50	43.48	78.90
100	45.01	77.37
200	46.21	79.51
400	46.71	85.63
800	47.26	85.93
1000	48.11	87.16
full dataset (62,000)	50.05	88.37

Table 3.4 Performance of algorithm with different numbers of clusters with SIFT features using 80,000 samples as test, 53,000 samples for cluster learning

No. of cluster	Performance(%) (feature)	Performance(%) (Block)
50	44.72	81.79
100	47.81	85.37
200	50.07	90.45
400	51.05	91.64
800	52.30	91.94
1000	52.75	93.13
full dataset (53,000)	54.27	93.73

3.7.3 Parameter comparison

In order to have a complete analysis of the methods, we briefly describe the parameters and their effect in each script identification method:

Table 3.5 Performance comparison of 3 script identification method (correct classification)

Algorithm	Performance(%)
Texture Method (Busch *et al.*, 2005)	87.2
shape codebook (Zhu *et al.*, 2009)	92.5
Proposed Method	97.6

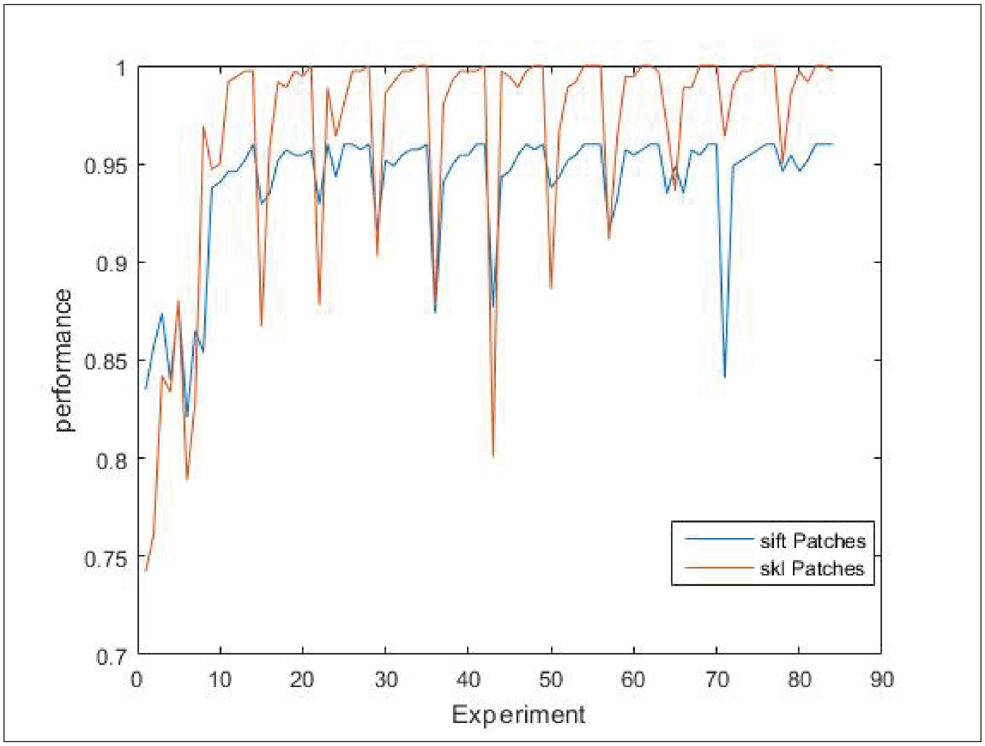


Figure 3.13 Comparison of performances of SIFT patches and Skeleton patches by different number of bases and clusters

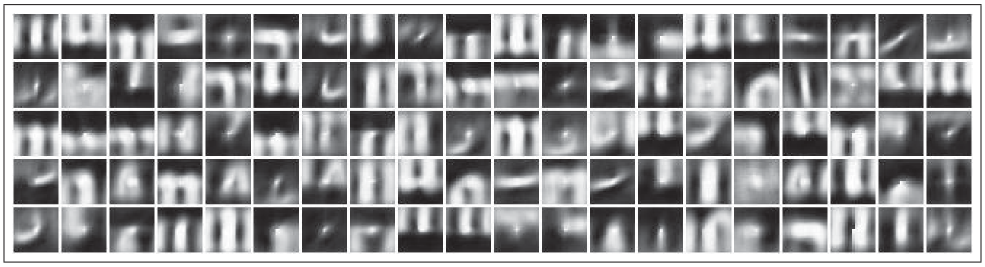


Figure 3.14 100 cluster centers obtained in low-dimensional space, shown in high-dimensional space

The proposed algorithm for script identification has two parameters that should be set by humans, number of bases in PNMF and number of clusters in K-means. The patch size will be set automatically according to measures from the document image. Our experiment shows that the algorithm is not sensitive to the size of the patch and that a 5-10 percent error in estimation of patch size does not have a significant effect on performance. Also, changing in the center point of a patch by 2-5 percent of the size of the patch is compensated for by the algorithm.

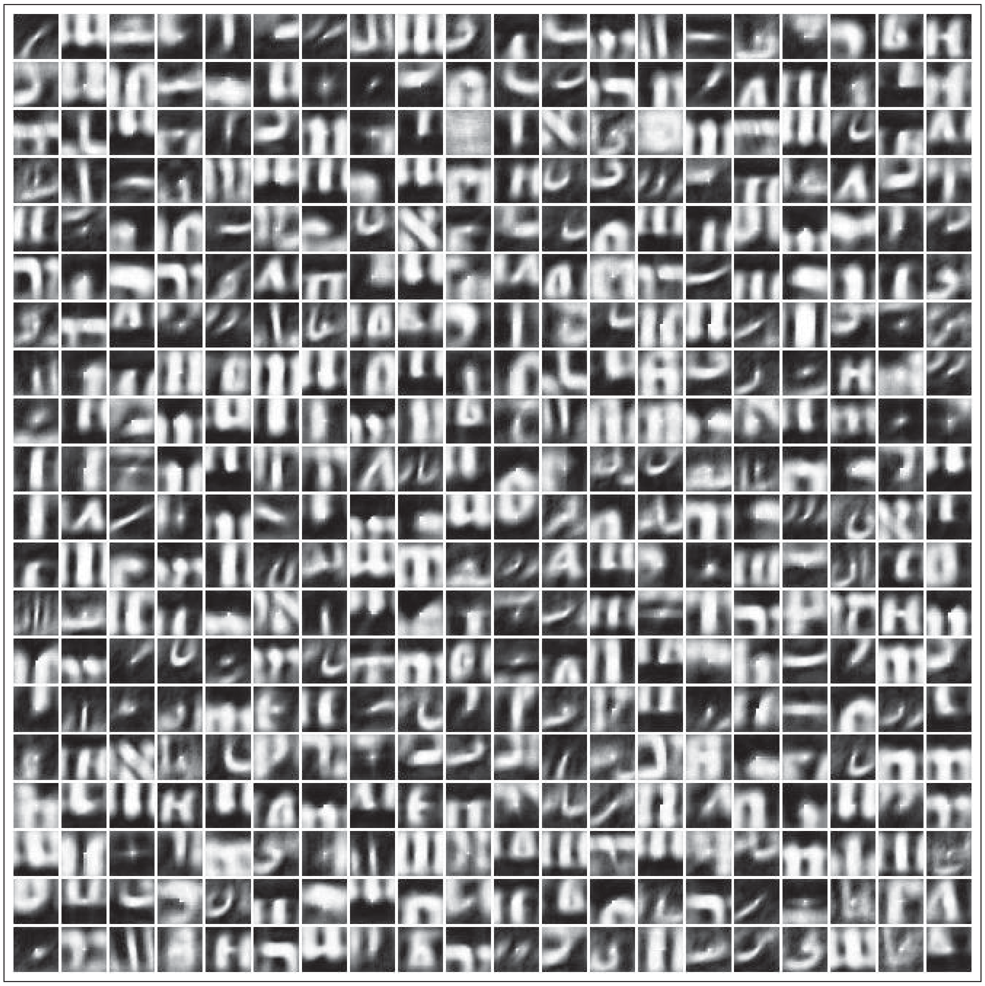


Figure 3.15 400 cluster centers obtained in low-dimensional space, shown in high-dimensional space

The method based on texture (Busch *et al.*, 2005) has several parameters that should be set correctly to achieve good performance, type of wavelet, number of levels in wavelet transform and the parameters used in coefficient quantization. The parameters of GLCM should also be set accordingly in order to get good features and results. They used K-means for clustering data to compensate for the effect of different fonts. The data are clustered by K-means where number of clusters is determined by the experiment. This method needs a great deal of normalization.

The method that uses shape codebook Zhu *et al.* (2009) is based on TAS (Three Adjacent Segments) features. The first parameter is a threshold that should be set for approximating lines from a set of curves; this step is highly resolution dependent. The similarity measure used is weighted measure between orientation similarity and length similarity of the TAS. Consequently, these weights are to be set properly in order to get an accurate similarity measure. The number of clusters for creating the shape codebook is the other parameter that has to be set in this method.

3.7.4 Robustness against noise

Robustness against noise is analyzed under two aspects: the way the patches are extracted and the feature extraction by NMF. The effect of noise on patch extraction is shown in Figure 3.16, where the top image is the original, the second image contains Gaussian noise, the third image is the binarization of the noisy image and, the fourth image is the image cleaned by removing the small components (individual pixels and very small components). In column a) the level of noise is low, and in column b) noise with higher levels is added to the image. We can see that the effect of noise is considerably reduced by this step even when the level of the noise is high. The NMF technique is well known in many fields and used for source separation and denoising. NMF processes the patches in a collaborative manner, extracts common information from the data and suppresses the noise. The patches extracted from document images contain common information about the word or character, and NMF uses it to retrieve the correct information, recover the missing parts of the characters and eliminate the noise. In Figure 3.17 some noisy patches and reconstruction with a different number of bases are shown. If we use a higher number of bases, we still have the noisy behaviors of the pixels in the patches; but using a reasonable number of bases for representation reduces that behavior, recovers the missing parts and eliminates the noise. This is one of the reasons why the best performance is obtained by using 200 bases in PNMF.

Texture methods are well known for robustness against noise and degradation because of their global features. The dataset that is created here consists of ancient manuscripts that contain noise and degradation. Comparing the performance of the proposed algorithm to the state-of-

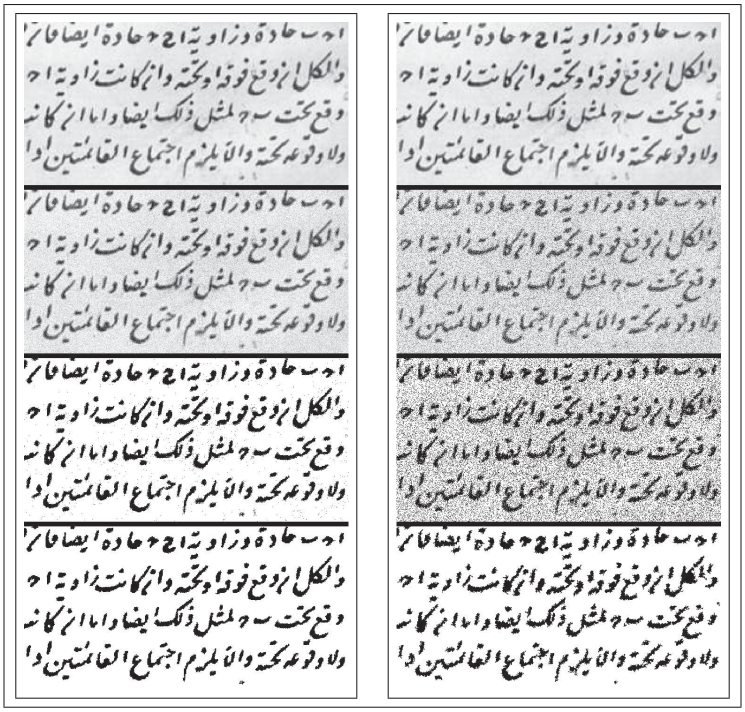


Figure 3.16 Pre-processing of image for keypoint extraction in the presence of noise, left) lower level of noise, right) higher level of noise

the-art and especially the texture method confirm the robustness of the proposed work against noise.

3.8 Conclusion

In this paper a novel method for script identification of ancient manuscripts based on image patches is proposed. This method does not need intensive normalization, and unlike the structure-based method, is very robust against noise and degradation. The proposed method is flexible for different levels of identification, and the experiments show the robustness and reliability of this method. In future work we will investigate better methods for extracting patches as well as other variants of NMF to improve the performance of script identification. We will also investigate how to generalize the proposed method to multi-script documents in which each text line contains different scripts.

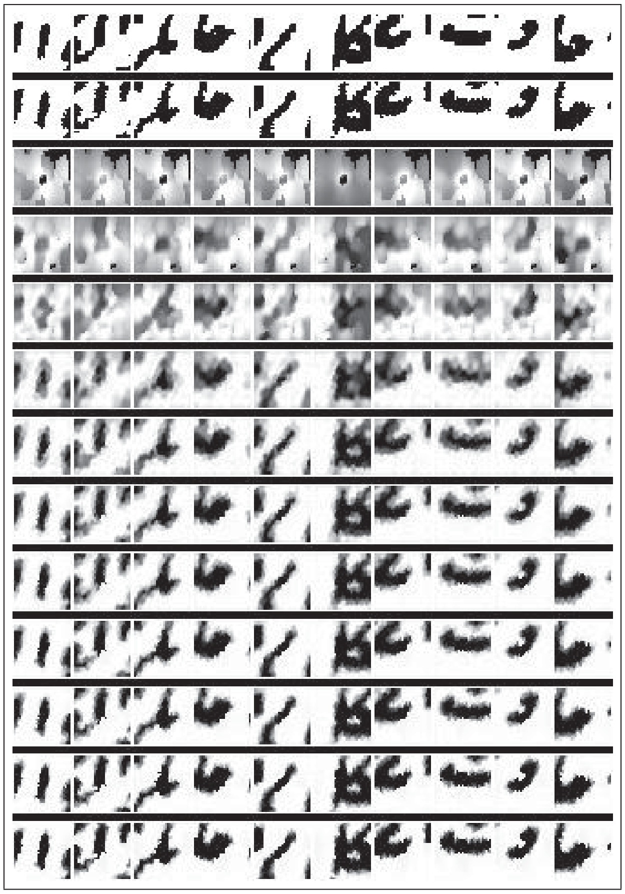


Figure 3.17 Representation and reconstruction of noisy patches with different number of bases

Acknowledgment

The authors would like to thank NSERC of Canada for their financial support.

CHAPTER 4

ARTICLE II - PFSI: PATCH-BASED ARABIC SCRIPT FONT AND STYLE IDENTIFICATION USING NON-NEGATIVE MATRIX FACTORIZATION

Ehsan Arabnejad[1], Mohamed Cheriet[1]

[1] Synchromedia Laboratory, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

Submitted to Elsevier Pattern Recognition

Abstract

Many studies in document understanding field showed that the Optical Character Recognition (OCR) engine trained on many fonts or styles does not perform well and also it has been proven that this is more challenging for Arabic script due to cursive nature of this script and lots of variation in fonts or styles. The aim of using Optical Font Recognition (OFR) is selecting OCR engine based on type of font that used in document image. Although, many methods were proposed for font identification, most of the methods are based on the assumption that the layout of the document image is known. This assumption is not true in many real situations. In this paper we proposed a method for font or style identification of document images written or printed in Arabic based on patch representation and Non-Negative Matrix Factorization (NMF). The proposed method does not need complete information about the layout and can be applied to different levels of layouts (from text line to paragraph). Each document image is represented by a series of patches that are extracted using skeleton map. These patches are processed by Non-Negative Matrix Tri-Factorization to learn dictionaries to be used for classification. We tested the proposed method on 2 datasets (1) Arabic printed dataset (ALPH-REGIM) composed of 7 fonts which is used for font identification and (2) database of ancient Arabic manuscripts with different writing styles which is used for style identification. The results of identification for these two databases show the robustness and the generalization power of the proposed method for different types of document image.

Keywords

Font Identification, Patch representation, Non-negative Matrix Factorization, clustering

4.1 Introduction

Writing systems use graphical shapes and symbols to represent elements (alphabets). These elements are evolved in time and their shapes are altered to create various writing styles which then are used in different situations and for different reasons such fast writing, artistic purposes, and etc. Some scripts such as Arabic are used in many countries with different languages and cultures so the elements faced many changes that appeared in various writing styles. By increasing the demands for publication and also usage of digital documents, different fonts are designed and used for publishing and printing books, magazines and etc..

Although, most of the recently produced materials are digital, there are many sources of information that are not in digital forms such as ancient manuscripts, old publications and etc. By increasing in the demand for digital world, many characters or word recognition systems have been proposed and developed to convert document image into digital but all of these systems are limited to working with specific fonts or styles. In order to improve the performance, OCR systems are trained on specific font or style and then an expert/user determines the proper OCR system based on font or style of a target document image. This is not practical if the goal is full automation of document understanding in big libraries.

With the help of Optical Font Recognition (OFR) the automation of recognition task (OCR engine) is improved by automatic selection of OCR engine based on fonts. This is also helpful for indexing document images in big libraries. Having information about the fonts or styles helps in better reproducing of digital materials and better understanding of cultural influence. The first step of OFR is extracting features from document image and the second step is using those features for classification.

Font identification methods can be divided into two categories based on type of method:

- Methods based on global features, these methods are text independent and skip information about the letters and the features describe global properties of document image.
- Methods based on local features, these methods are based on analyzing the individual object (letters). In these methods the components of document image (letters) are isolated and then based on similarity from textual point of view they differentiate between different fonts.

The font identification methods can also be classified into 4 groups, based on type of features:

- texture features such as method in (Dhandra *et al.*, 2006)
- typographical features such as method in (Busch *et al.*, 2005)
- structural features such as method in (Aithal *et al.*, 2010)
- automatic learning of features such as method in (Roy *et al.*, 2010)

One of the main limitations of most font identification methods is that the level of layout should be known for extracting accurate features. Another drawback is that a lot of information about the font is lost in feature extraction step due to manual designing of features. From another point of view, the opinion of an expert is needed to design new features for new fonts so adding new fonts needs verification or modification of current features or introduction of new features.

In this work, we propose a novel method for font and style identification of printed and ancient manuscripts based on patch representation and Non-Negative Matrix Factorization(NMF). This method is robust against degradation and noise and is flexible to be used with different levels of layout.

The patch representation allows for modeling of the complex structure of document image by simple structure of the patches. The proposed method skips the need for having information about the layout of the page. This also helps us to avoid intensive normalization steps that are needed for global-based methods and provides a way to use all of information about the font in any level of layout for feature extraction and font classification. The features are extracted in automatic way so it can be easily generalized to any new type of font.

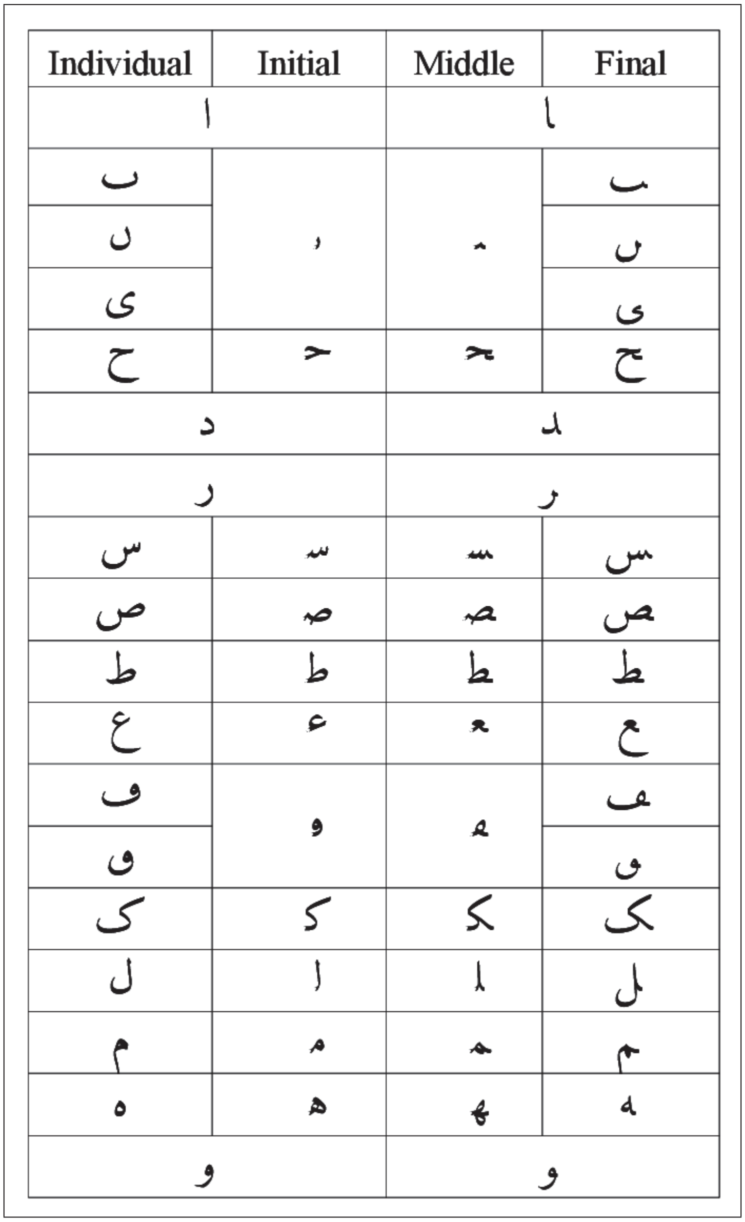
The organization of this paper is as follows. In section 4.2 a brief description of Arabic script and fonts are brought. In section 4.3 related state of the art methods are discussed. The motivation is discussed in section 4.4 and the proposed method are presented in section 4.5. The experiments and evaluation are presented in section 4.6 and the summary of the work is presented in the conclusion.

4.2 Arabic Script

Arabic script is composed of 28 letters and by considering the shapes of the letters (in individual form) and skipping the dots, it has only 16 distinctive letters. This script is used for writing in many other languages such as Farsi and Urdu so some additional letters were introduced to adapt this script to those languages. In most of these situations the new letter is introduced by only adding, moving or removing dots of existing letters.

Arabic script is cursive and the letters are attached to each other to create words and the shapes of the letters should be altered according to the position in the word, preceding and succeeding letters. Many letters have 4 different shapes which are called initial, middle, final and individual forms and a few of them have only two forms. Some of the letters are only different in position or number of dots which is a very small component compared to the size of the letters and also it might be placed differently especially in handwritings. By considering just the shapes of the letters, Arabic script has 58 different shapes that can be seen in Figure 4.1. We can see that many of the letters are composed of two parts, a common part that is appeared in 4(2) different forms and a part which mostly round or horizontal stroke used in individual and/or final forms.

Arabic script is written in many different styles and many different fonts are designed for publishing in Arabic. Some of famous and most used Arabic fonts are shown in Figure 4.2. As we can see from this figure, the shapes of the letters, connections between them and distance between words are different in different fonts.



Individual	Initial	Middle	Final
ا		ـا	
ٮ	ٮـ	ـٮـ	ـٮ
ں			ـں
ى			ـى
ح	حـ	ـحـ	ـح
د		ـد	
ر		ـر	
س	سـ	ـسـ	ـس
ص	صـ	ـصـ	ـص
ط	طـ	ـطـ	ـط
ع	عـ	ـعـ	ـع
ڡ	ڡـ	ـڡـ	ـڡ
ٯ			ـٯ
ک	کـ	ـکـ	ـک
ل	لـ	ـلـ	ـل
م	مـ	ـمـ	ـم
ه	هـ	ـهـ	ـه
و		ـو	

Figure 4.1 Different shapes of Letters used in Arabic script

4.3 Related Work

In this section we briefly introduce some leading and pioneering methods for font identification of document images.

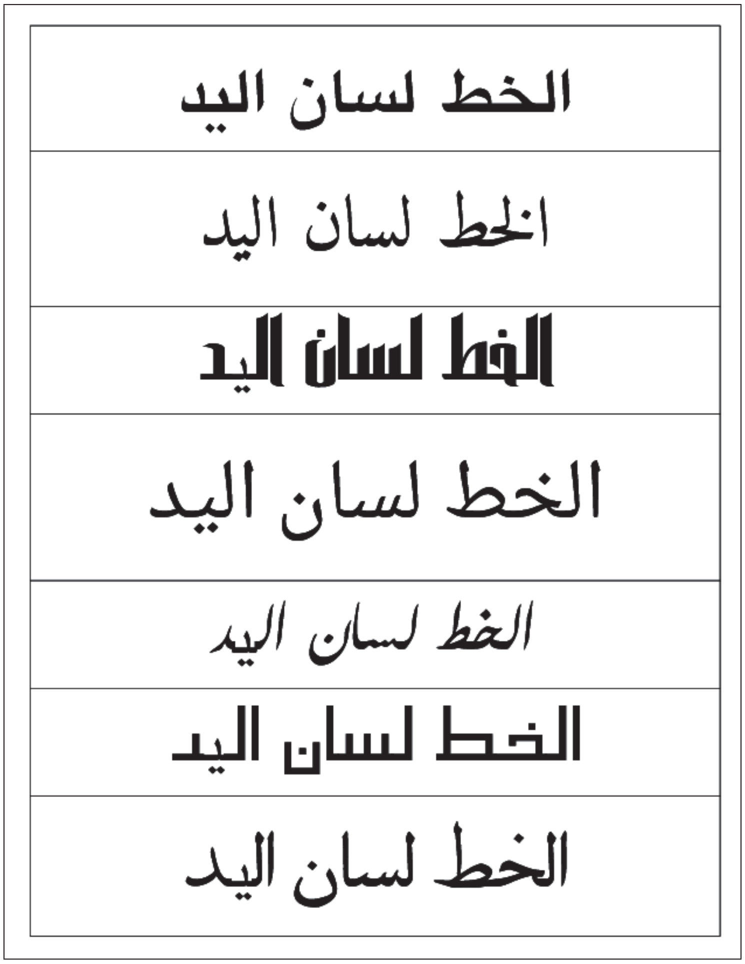


Figure 4.2 Some examples of Arabic font

In Zramdini & Ingold (1998) the typographical features of text are used for font identification. They used some global features that capture the property of the text and some local features obtained from the connected components for font identification. In Schreyer *et al.* (1999) the authors defined and introduced the notion of Texton and then used it for font identification. In Zhu *et al.* (2001) the concept of texture features is used for font identification. The authors proposed to use mean and standard deviation of the output of Gabor filter applied in many scales and direction as features for font identification. In Fang *et al.* (2002) the authors proposed a method for font identification of individual characters based on texture features. They used rule based decision in 6 levels for font identification. In Lee *et al.* (2003) a method for font identification of individual character based on NMF is proposed. In their method NMF is used to learn features from the fonts and then nearest neighbor classifier is used for classification. In Zhang *et al.* (2004) the authors proposed a method for separating italic font by analyzing stroke-pattern. Their method is based on wavelet transform and features are properties of vertical and diagonal strokes. In Ha *et al.* (2005) the method for font identification based on optimized directional Gabor filters is proposed. These filters optimized by genetic algorithm for the each font specifically.

In Sun (2006) a method for font identification based on stroke template is proposed. The authors proposed a procedure for extracting stroke templates and then these stroke are classified and used for font identification using Bayes's decision rule. In Ramanathan *et al.* (2009b) a method for font identification using Gabor filters and SVM is proposed. This method is applied on the image that divided into 9 overlapping areas and features are the mean and standard deviation of Gabor filters in many directions. In Chawki & Labiba (2010) a method for Arabic font identification based on GLCM is proposed. GLCM features are extracted from binary image in different distances and different directions. Some of already known features based on texture and also new features based on Gray Level Run Length matrix is used for classification. In Slimane *et al.* (2010) the method based on Gaussian mixture model is proposed for font identification. They extracted features in two steps, first by sliding window with the width of 4 pixels and extracting some features and second by horizontal and vertical projection. They used GMM to model features of each font and the classification is based maximum probability.

In Khosravi & Kabir (2010) method based on gradient features for font identification is proposed. The image blocks are divided into some sub-blocks and then features based gradients are extracted. These features are calculated using the Sobel gradient.

In Pourasad *et al.* (2011) a method for Farsi font recognition based on spatial matching is proposed that works on isolated letters of different Farsi fonts. The authors proposed to extracted contour points in a random way and gradient is calculated for each point. They used Euclidean distance for matching and in order to speed up the algorithm , they used gradient and distance intervals and the descriptor (called Spatial Gradient Difference Descriptor) for comparison between different fonts.

In Bataineh *et al.* (2011) the method for Arabic calligraphy classification is proposed. The features are based on Edge Direction Matrix. First order relation and second order relation of pixels are obtained and some features such as correlation and homogeneity is used to prepare features and then neural network is used for classification. In Chanda (2012) the authors proposed a method for font identification of Indic script based on curvature features. These features are extracted in different directions, scale and steps and SVM is used for classification. In Ben Moussa *et al.* (2010) a method for font identification of Arabic script based on fractal dimension for feature extraction and neural network for classification is proposed.

4.4 Motivation

In this paper, we address the problem of font and style identification in printed/ancient manuscripts by proposing a novel, robust and flexible method which is based on patch representation. Image patches, defined as local blocks of images, are at the core of various image processing applications. This framework has many usages in image and video compression, image restoration/reconstruction and has been used recently for script identification in Arabnejad *et al.* (2017) and Gomez *et al.* (2017).

Image patches are very convenient to use for different reasons. They provide the local properties of an image and many assumptions about the image can be considered valid in local level such low-color variation, uniformity of texture and affinity of transforms. Using image patches helps in simplifying most of the assumptions about nature of the images.

From another point of view, images and especially document images contain highly redundant information and have complex structure but if we look at them at the patch level we can see that they are composed of simple components. The redundancy can be used to extract or restore the similar patches and the complex structure of images can be modeled simply and effectively by simple structure of the patches.

A document image, despite of having a highly complex structure and layout, is composed of a set of limited object called letters. These letters have specific shape and sequence of them create the words and the sentences. For some scripts the shape of the letters do not change by position but for some scripts, such as Arabic, the shapes of the letters change based on position in a word, the preceding and/or succeeding letters.

For scripts that are composed of isolated letters, these letters can be separated from the text by performing layout analysis and measuring the constraints that are used to write in that script. For naturally cursive script this segmentation is very difficult as the boundaries of the characters are not well defined.

Patch based representation for the document images that are composed of cursive scripts helps in simplifying the problem. So if the patches are selected correctly, they can be part of a letter, a specific letter or combination of some letters so that will be a good representation for the elements of those scripts. These patches can be processed by proper methods for restoration, feature extraction, classification and etc.

If we look at the state of the art methods for font or style identification, we see that many limitations and drawbacks are not addressed. The methods based on global features (mostly based on texture features) need many pre-processing steps in order to prepare (normalize) document image for feature extraction and classification. In most of the methods, the models are built based on 100% accurate layout analysis which is not correct in many cases. Dealing with the specific level of layout is another limitation of these methods so if the layout of the target image changes (for example different number of text lines) then these type of methods cannot be applied. In most of the local based methods, the label (textual information) of the letters should be known as prior information so the practical usages of these methods are under question. In many of the font identification methods an expert observes data and designs specific features for separating different fonts so adding new fonts needs the opinion of experts for evaluating or modifying the designed features or designing new features.

Our method is based on the assumption that the patches carry enough information about the font or style and this information can be extracted automatically and used for classification of document images based on different fonts or styles. The proposed method has many advantages compared to other methods. By using patch representation we avoid segmentation of document image which is challenging problem by itself. We do not need many samples for training and extracting features so only few document images can be used to extract patches for learning and classification. Other advantages of the proposed method is an automatic learning of features that does not need opinion of experts and these features can be easily extracted from new fonts. Another advantage is that the constraint about the layout is skipped and the patches extracted at any level of layout can be combined with other levels of layout for classification.

4.4.1 Non-Negative Matrix Factorization (NMF)

4.4.1.1 Background

One of the main ideas behind the approaches for noise removal is replacing data by its lower-dimensional subspace approximation. There are many data available in various fields that exhibit the non-negative property (such as text and image data) and any process on this type of data is desired to maintain this property. Non-Negative Matrix Factorization (NMF) introduced in Lee & Seung (1999) by this aim and is used for finding low-rank approximation of matrix or extracting features with non-negativity constraint.

The generic form of Non-Negative Matrix Factorization (NMF) is as follows: Given a non-negative matrix (data matrix) $X \in R^{m \times n}$ and a positive integer $k < min(m,n)$, the goal is to find non-negative matrices $H \in R^{m \times k}$ and $W \in R^{k \times n}$ such that:

$$X = HW \tag{4.1}$$

Finding the exact solution for NMF is not feasible but the approximation with the rank k can be obtained. One of the main steps of NMF is defining a similarity metric that will be used, as objective function, to measure the quality of approximation. The most common similarity measures are Euclidean distance (EUD) used in Lee & Seung (1999) and Kullback-Leibler divergence (KLD) which is introduced in Kullback & Leibler (1951) and used in Lee & Seung (1999). Although, the original NMF is used successfully in many applications it has some drawbacks so it is modified in different ways to be adapted to different problems.

It has been observed that different types of data have different behaviors so in order to accurately measure the similarity, various similarity measures are used: such as the Minkowsky family of metrics or l_p-norm, earth mover's distance Sandler & Lindenbaum (2011), $\alpha-$ divergence Cichocki *et al.* (2008), $\beta-$ divergence Kompass (2007), $\gamma-$ divergence Cichocki *et al.* (2006), Bergman distance Dhillon & Sra (2005), and $\alpha-\beta-$ divergence Cichocki *et al.* (2011).

The original NMF is modified by adding various constraints to get stationary solution(as much as possible) such as: sparsity constrains in Hoyer (2002), orthogonality constraints of bases in Choi (2008), discriminant constraints in Wang & Jia (2004), and manifold regularity constraints in Cai *et al.* (2011).

Also the quality of approximation is investigated in some fields by modifying the factorization : weighed NMF in Kim & Choi (2009), convolutive NMF in Smaragdis (2007) and Non-Negative Matrix Tri-Factorization in Yoo & Choi (2010).

In some methods such as Convex Matrix Tri-factorization in Ding *et al.* (2010), Non-Negative Tensor Factorization in Shashua & Hazan (2005), Non-Negative Matrix-set Factorization in Li & Zhang (2007) and Kernel-NMF in Lin (2007) the non-negativity constraints on data are removed and the NMF is adapted to other types of data.

The objective function of original NMF is not convex for two variables but it is convex for each variable. The most well-known form of NMF is introduced by Lee and Seung in Lee & Seung (1999) as the authors proposed the multiplicative update rules to solve NMF:

$$W_{kj}^{(t+1)} = W_{kj}^{(t)} \frac{(H^T X)_{kj}}{(H^T H W^{(t)})_{kj}} \tag{4.2}$$

$$H_{ik}^{(t+1)} = H_{ik}^{(t)} \frac{(X W^T)_{ik}}{(H^{(t)} W^T W)_{ik}} \tag{4.3}$$

where t is the iteration index, i row index and k column index.

4.5 Patch-based Font and Style Identification Using NMF

The proposed method for font or style identification is illustrated in Figure 4.3. After pre-processing and preparing the data, several patches are extracted from the images and then NMF is used to learn bases from this set of patches. This process repeated for each font or style in the dataset. For the test set, a similar process is used for extracting the patches and the bases that are learned in the training step is used to represent the patches of test set. Reconstruction error is used as a criterion for classification of different fonts.

4.5.1 Pre-processing

Pre-processing steps that are used in this paper are binarization and skew correction. In the binarization step a document image is converted to black and white, foreground pixels appear as black and background pixels as white. There are many advanced methods for binarization of document images such as Nafchi *et al.* (2014) and Howe (2013). In this paper we used simple global threshold based on Otsu's method Otsu (1979) for binarization of document images. Skew in document image is defined as the angle of the baselines with horizontal line. In order to remove the possible skew of document image, we used variance of horizontal projection in different angles as a measure to estimate and correct the skew of document image. This is based

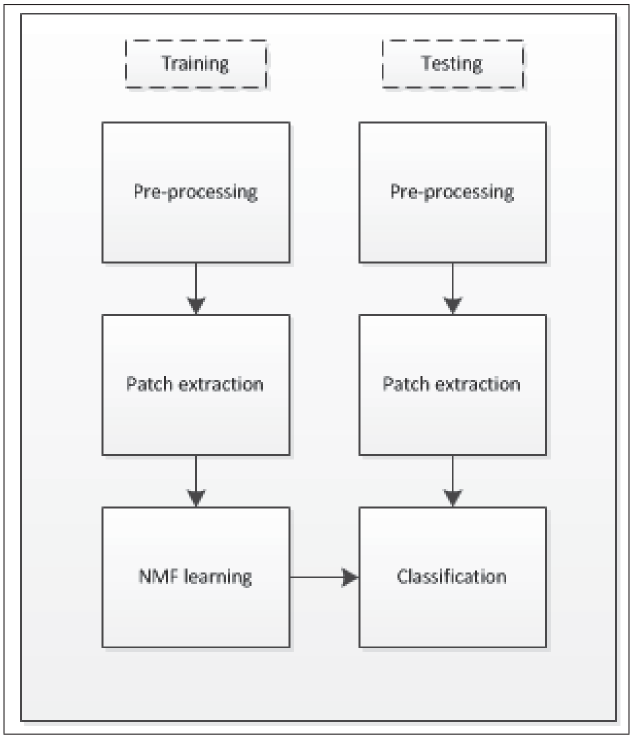


Figure 4.3 Flowchart of proposed method

on the observation that horizontal text-lines produce more variation in horizontal projection than the skewed text lines.

Document images (printed/ancient) are degraded by noise and degradation, which creates many unwanted patches. Some letters of Arabic script have dots and in some cases diacritics are used for better legibility or artistic purposes. These components do not carry valuable information and should be removed. Individual pixels and small components are removed by analyzing the area of those components and using a threshold based on average size.

4.5.2 Patch extraction

The goal of the patch-based method is to represent an image with a set of overlapping patches. Different representations can be obtained by setting different values for the center of the patches and size of the patches. There are many approaches for detecting or determining the keypoints that will be used as the center of the patches such as regular grid, random sampling and methods based on keypoint extraction such as Bay *et al.* (2008). The latter methods are based on detecting corner points that are robust to rotation, translation and/or other transforms.

Document images are highly structured and many pixels (80-90 percent of pixels) belong to background and the patches extracted from background do not have significant information. In order to reduce number of patches, the center points of the patches are selected from the foreground pixels(text).

Extracting all of the patches from foreground pixel is not needed because many patches are the translation of another patch/patches by one pixel and the changes are not significant. In order to remove these kinds of patches, we used the approach introduced in Farrahi Moghaddam *et al.* (2012) that uses the skeleton map of the connected component as the reference for the center of the patches. In Figure 4.4 skeleton map and patches of a sample image are shown.

We have followed two approaches for patch size estimation which depends on type of document image. For machine printed document images, we used global measures of text height, distance between text lines and stroke width for determining the size of the patch. These measures can be obtained accurately as the layout of these type of images do not change.

Unlike machine-printed document images, the ancient manuscripts have highly unconstrained layout, so the measures mentioned above cannot be calculated accurately. For the training set we manually set the patch size for each image and then we extract the patches and use NMF for learning bases. For the test set, we use the estimation of patch size by the method introduced above (for machine printed documents). We set different values for the patch size based on the estimated size, and then we extract patches accordingly and do the representation and reconstruction. The representation that has the minimum reconstruction error is selected as final representation. This is based on the fact that bases obtained in the training step is calculated for specific text height and if the patches of the new image are obtained with wrong text height the representation error is higher compared to the situation where the text height is estimated correctly.

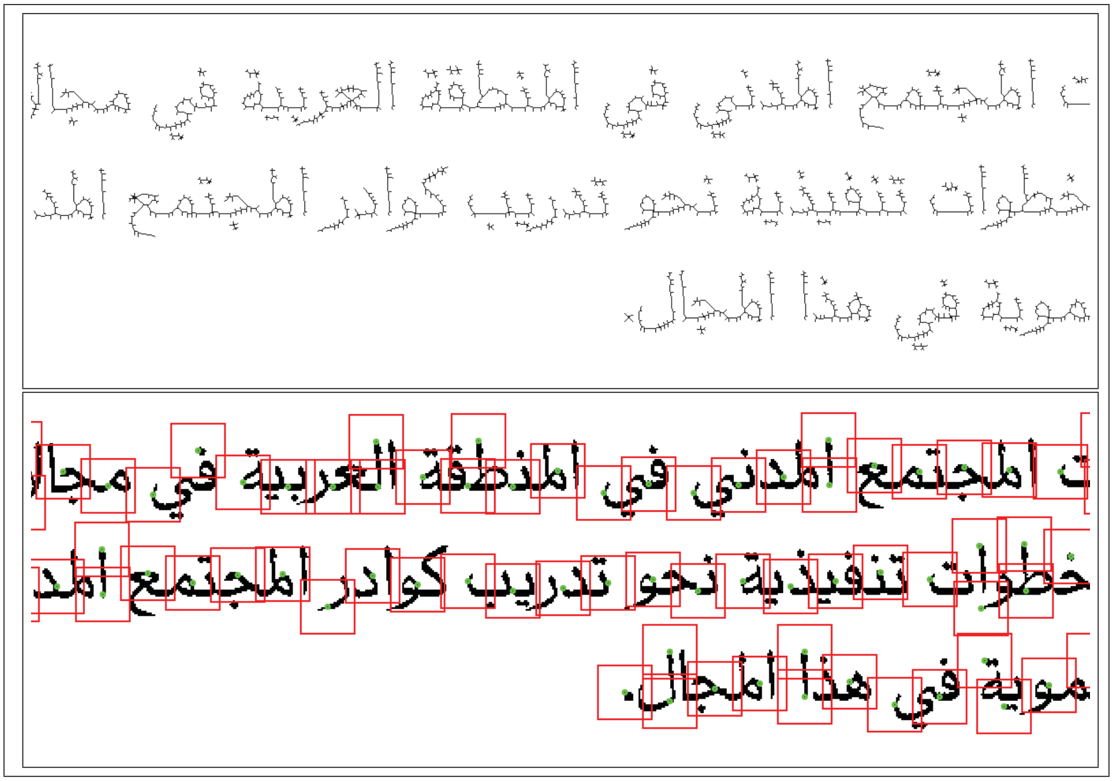

ت المجتمع المدني في المنطقة العربية في مجال
خطوات تنفيذية نحو تدريب كوادر المجتمع المد
موية في هذا المجال.

Figure 4.4 top) Skeleton map, bottom) sample keypoints and the corresponding patches

4.5.3 Feature extraction

In patch-based algorithms information is extracted from finite dimensional windows, a subset of the pixels within an image. Document images are highly structured and these subsets of pixels contain part of a character, a character or a combination of several characters. Different levels can be obtained by changing the size of the patch.

One specification of NMF is non-negativity constraints that produce part-based representation. Another useful property of NMF is collaborative learning. These two properties provide a proper framework for representation of noisy and degraded image patches. Part based representation provides an abstract representation and collaborative learning allows many of patches contribute in learning step so if some parts of the characters corrupted by degradation or a noise the collaborative learning allows for retrieving the correct information.

The proposed method for font and style identification is based on Non-Negative Matrix Trifactorization. This method belongs to the family bi-clustering method that cluster rows and columns of the matrix of data simultaneously. Two parameters that determine the behavior of NMTF are the number of bases and the number of clusters. The NMTF algorithm is as follows:

$$X = FSG^T \tag{4.4}$$

Depending on the composition of X, these matrices can be interpreted differently. If X is a matrix with the rows as features and the columns as samples, F can be considered as bases matrix in high-dimensional space, then G is a coefficient matrix and S is a bases matrix in the low-dimensional space.

If we consider Euclidean distance as the similarity measure between the data and representation, the objective function for NMTF is as follows:

$$\min_{F>=0,S>=0,G>=0} \left\| X - FSG^T \right\|_2^F \tag{4.5}$$

The objective function 4.5 is minimized with two constraints, orthogonality of bases and orthogonality of coefficients in Ding *et al.* (2006). The first constraint forces the algorithm to learn local bases which improve robustness of representation against noise and degradation. The second constraint forces the algorithm to learn very sparse coefficient matrix that can be interpreted as cluster index.

The multiplicative update rules for 3 matrices according to Ding *et al.* (2006) are as follows:

$$F_{ij} = F_{ij} \frac{\left(XGS^T\right)_{ij}}{\left(FF^T XGS^T\right)_{ij}} \tag{4.6}$$

$$S_{ij} = S_{ij}\frac{\left(F^T XG\right)_{ij}}{(F^T FSG^T G)_{ij}} \tag{4.7}$$

$$G_{ij} = G_{ij}\frac{\left(X^T FS\right)_{ij}}{(GG^T X^T FS)_{ij}} \tag{4.8}$$

where i is row index and j is column index. For these 3 matrices i and j change according to the size of the matrices. As the updating rules above are not stable, the modified updating rules in Ding *et al.* (2006) are as follows:

$$F_{ij} = F_{ij}\sqrt{\frac{(XGS^T)_{ij}}{(FF^T XGS^T)_{ij}}} \tag{4.9}$$

$$S_{ij} = S_{ij}\sqrt{\frac{(F^T XG)_{ij}}{(F^T FSG^T G)_{ij}}} \tag{4.10}$$

$$G_{ij} = G_{ij}\sqrt{\frac{(X^T FS)_{ij}}{(GG^T X^T FS)_{ij}}} \tag{4.11}$$

4.5.4 Classification

In the proposed method for font or style identification, the classification is done based on minimum reconstruction error. As it is mentioned before, we learn dictionaries for each font/style separately using NMTF so we have bases matrix G and cluster matrix S for each set of fonts. For each image in the test set we follow the same procedure for pre-processing and patch extraction. The extracted patches along with S and G matrices are used to obtain the coefficient matrix. After obtaining the coefficients matrix F, we calculate the error between the original samples and their representations and we repeat this for all of the fonts or styles in the dataset.

The label of the font or style set that produce lower reconstruction error is selected as label for that set of patches and that image.

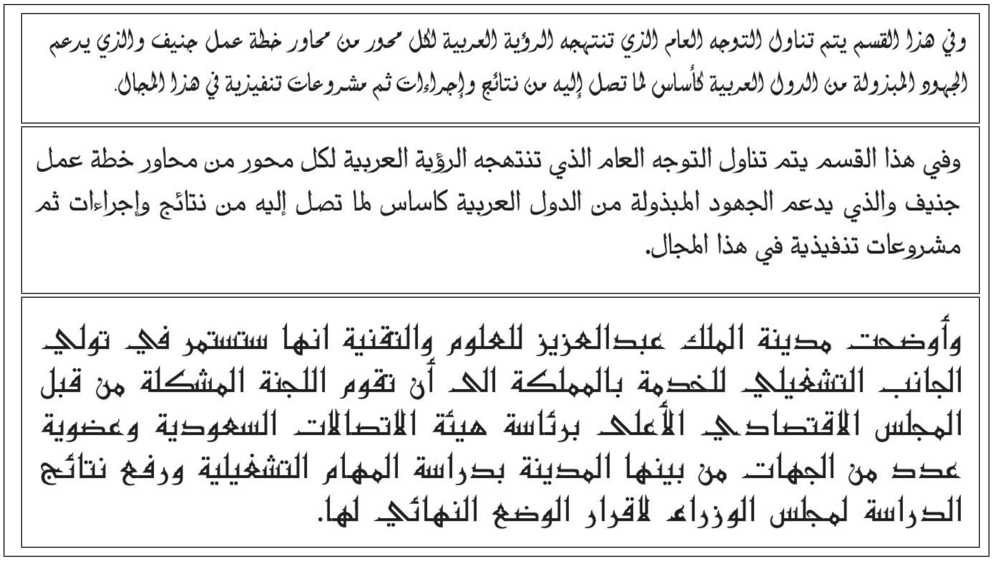
4.6 Experimental Results

The proposed font and style identification method is evaluated on two datasets. Different experiments are performed to show the performance and robustness of the proposed method.

4.6.1 Database

In order to validate the performance of the proposed method we used two datasets, a dataset of machine-printed and a dataset of ancient manuscripts. As machine-printed dataset, we used a subset of ALPH-REGIM dataset Ben Moussa *et al.* (2008) that contained 1500 images in 7 different fonts (we used the Arabic section). The images are in different number of text lines (2 to 15 text lines) for different fonts. Few examples of images of this dataset are shown in Figure 4.5. The images are created with one scale so in order to see the effect of patch size estimation, we randomly change the size of the images by the scale of 0.5 to 1.5 and tested the proposed method. Also the images are skewed randomly from -3 to 3 degrees in order to see the effect of the skew correction in the performance.

As an ancient manuscript dataset, we created a dataset Arabic script document images written in different styles. We collected 500 document images from different libraries. These images are written in different styles such as Naskh, Tholoth, Nastaliq and Koufi. These styles are some of the main styles used for writing Arabic. Some examples of this dataset are shown in Figure 4.6. To prepare this dataset for the experiments, we randomly select from 2 to 10 text lines from these images. The text lines are selected so that they cover the width of the page layout. These images are in different sizes, resolution and layouts. There are around 1200 samples created from this set of documents, some pages contain many text lines so more examples created from those documents with the aim that two samples do not have overlap and do not cover common part of texts.



وفي هذا القسم يتم تناول التوجه العام الذي تنتهجه الرؤية العربية لكل محور من محاور خطة عمل جنيف والذي يدعم
الجهود المبذولة من الدول العربية كأساس لما تصل إليه من نتائج وإجراءات ثم مشروعات تنفيذية في هذا المجال.

وفي هذا القسم يتم تناول التوجه العام الذي تنتهجه الرؤية العربية لكل محور من محاور خطة عمل
جنيف والذي يدعم الجهود المبذولة من الدول العربية كاساس لما تصل إليه من نتائج وإجراءات ثم
مشروعات تنفيذية في هذا المجال.

وأوضحت مدينة الملك عبدالعزيز للعلوم والتقنية انها ستستمر في تولي
الجانب التشغيلي للخدمة بالمملكة الى أن تقوم اللجنة المشكلة من قبل
المجلس الاقتصادي الأعلى برئاسة هيئة الاتصالات السعودية وعضوية
عدد من الجهات من بينها المدينة بدراسة المهام التشغيلية ورفع نتائج
الدراسة لمجلس الوزراء لاقرار الوضع النهائي لها.

Figure 4.5 Examples of few different Arabic fonts in ALPH-REGIM dataset

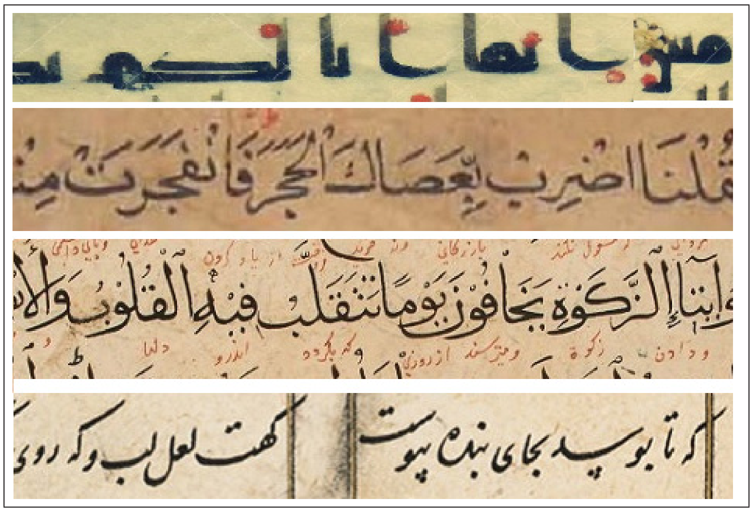

قلنا اضرب بعصاك الحجر فانفجرت من
إيتاء الزكوة يخافون يوما تتقلب فيه القلوب والأ

Figure 4.6 Examples of different Arabic handwriting styles

4.6.2 Experimental Results

In this section the result of the proposed method for printed/ancient manuscript is shown.

4.6.3 Font Identification

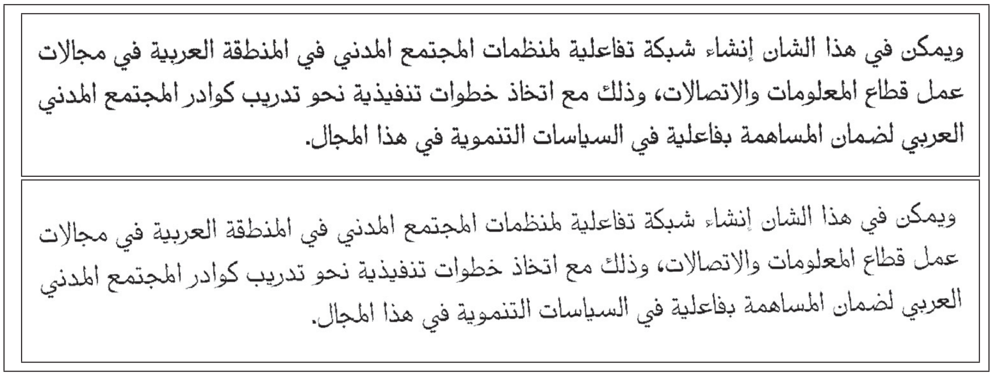
To test the proposed method, we divided the ALPH-REGIM dataset into training and testing set. As the lowest level of information are the patches and our method does not need many samples, we randomly selected 5% of images for training and the rest for testing. We extracted the patches of images of training set with the procedure mentioned before. This leads to the 85,000-127,000 f or each class of fonts.

Two parameters of representation by NMTF are l, number of bases in original space and k, number of bases in low-dimensional space. We tested different values for k and l. The best parameters based on convergence time, reconstruction error and quality of representation are obtained as 200 and 500. These set of parameters are used to learn two matrices of S and G for each font. One of the benefits of this procedure is that if a new type of font is added to the dataset, the training step will only be repeated for that font. The result of font identification on this dataset is shown in table 4.1. In this table and similar tables columns are true classes and rows are predictions.

Table 4.1 Confusion table of font identification (in percent)

fonts	Diwani	Hijaz	Kharj	Khoubar	Koufi	Naskh	Tholoth
Diwani	99.07	0	0	0	0	0	0
Hijaz	0	100.00	0	0	0	0	0
Kharj	0	0	100.00	0	0	4.85	0
Khoubar	0	0	0	100.00	0	0	0
Koufi	0	0	0	0	100.00	0	0
Naskh	0	0	0	0	0	95.14	0
Tholoth	0.9259	0	0	0	0	0	100.00

In order to test the robustness of the algorithm, we randomly applied transformation on the images. These transform are skew transform of -2 to 2 degree, scale transform of 0.5 to 1.5 and also perspective distortion. One example image and its transformed version (with random parameters) are shown in Figure 4.7. The result of font identification of these transformed images is shown in table 4.2 and we can see that the algorithm performed well in this experiment. The errors happen in the images that perspective transform changes the text size more than 15% of average patch size. The skew correction and patch size detection steps detect the skew and scale of the transformations accurately and these transformations do not have significant effect on the performance.



ويمكن في هذا الشان إنشاء شبكة تفاعلية لمنظمات المجتمع المدني في المنطقة العربية في مجالات
عمل قطاع المعلومات والاتصالات، وذلك مع اتخاذ خطوات تنفيذية نحو تدريب كوادر المجتمع المدني
العربي لضمان المساهمة بفاعلية في السياسات التنموية في هذا المجال.

ويمكن في هذا الشان إنشاء شبكة تفاعلية لمنظمات المجتمع المدني في المنطقة العربية في مجالات
عمل قطاع المعلومات والاتصالات، وذلك مع اتخاذ خطوات تنفيذية نحو تدريب كوادر المجتمع المدني
العربي لضمان المساهمة بفاعلية في السياسات التنموية في هذا المجال.

Figure 4.7 (top) Original image and (bottom) transformed image (perspective transform)

Table 4.2 Confusion table of font identification with the added transformations (in percent)

font	Diwani	Hijaz	Kharj	Khoubar	Koufi	Naskh	Tholoth
Diwani	93.69	0	0.86	0	0	0	0
Hijaz	0	93.40	0	0	2.83	0	0
Kharj	1.80	2.83	98.28	2.61	0	2.83	1.74
Khoubar	0	1.99	0	96.52	0	0	0
Koufi	0	0	0	0	97.17	0	0
Naskh	4.50	1.89	0.86	0.87	0	96.23	3.10
Tholoth	0	0	0	0	0	0.94	93.91

In the third experiment we tested the robustness of the algorithm against the error in parameters of patch extraction step, size of the patches and skew of the images. After skew correction and patch size estimation, the images are randomly skewed and scaled but the patches are extracted using the initially estimated parameters. The results of this experiment are shown in table 4.3.

Analyses of the results shows that for low levels of transformation, the algorithm is able to maintain the performance and accurately classify the image. Error in the scale estimation of the image by $\pm 10\%$ of original size does not have significant effect on the performance. Also skew errors of $\pm 1^\circ$ is compensated by the proposed patch representation and NMF.

For higher level of transforms, the performance of classification of Khoubar and Diwani fonts show more declines. That is because of the shape of the fonts and also similarity between those fonts. We can see that the skew transform has more effect on the performance than the scale transform. The representation of patches obtained by NMTF (FS) in the first stage is sparse and abstract so it is robust against scale transform. The proposed method performs well in low levels of distortion because NMTF representation is robust for up to 10% error in patch size estimation.

Table 4.3 Confusion table of Font identification with the higher level of added transformation (in percent)

font	Diwani	Hijaz	Kharj	Khoubar	Koufi	Naskh	Tholoth
Diwani	83.33	0	0	0	0	0	0
Hijaz	0	82.52	0	0	0	0	0
Kharj	3.70	2.91	100	0.89	7.76	11.65	0.89
Khoubar	0	1.94	0	57.14	0	0	0
Koufi	0	0	0	0	92.23	0	0
Naskh	12.96	12.62	0	16.96	0	88.35	8.92
Tholoth	0	0	0	25.01	0	0	90.17

4.6.4 Style Identification

As mentioned before, to validate the performance of the proposed method for style identification, we created dataset of ancient manuscripts and the proposed method is evaluated on this dataset. This dataset contains different types of documents, in different sizes, different layouts and with different number of text lines so it is a highly challenging dataset for style identification. We divided this dataset to training and testing set. Due to variation in styles and layout and presence of noise and degradation, we divided this dataset to 30% for the train set and 70% for the test set. For NMTF we used the same parameters for *l* and *k*. The result of style identification method on ancient manuscript is shown in table 4.4.

Table 4.4 Confusion table of style identification (in percent)

font	Naskh	Tholoth	Nastaliq	Koufi
Naskh	95.46	2.85	1.32	3.25
Tholoth	3.75	96.23	0	1.84
Nastaliq	0	0	98.68	0
Koufi	0.79	0	0	94.91

Analyzes of the results show that most of errors are incorrect classification of Naskh and Tholoth style. It can be seen from the Figure 4.5 that shapes of some letters are highly similar between these two styles. The best performance obtained for identification of Nastaliq style as this style has many distinctive features such as more curves, different type of connection between letters and also skewed baseline for each word. The error for this style are for images that have lots of noise and degradation and also error in patch size estimation.

The proposed method compared with method in Ben Moussa *et al.* (2010) for font/style identification and the results are shown in table 4.5. We can see that for font identification the difference between the performance of these two methods is not significant. While the method in Ben Moussa *et al.* (2010) performs better than proposed method for font identification, the proposed method has better performance in style identification. These results show the generalization power of the proposed method from font identification (printed document image) to style identification (handwriting).

4.7 Conclusion

In this paper we proposed a novel method for font and style identification of machine-printed and ancient manuscripts. The proposed method is flexible for any level of layout, robust against

Table 4.5 Accuracy comparison of font and style identification

Method	dataset	Accuracy
proposed method	(font)	99.42
Method in Ben Moussa *et al.* (2010)	(font)	99.89
proposed method	(Style)	94.57
Method in Ben Moussa *et al.* (2010)	(style)	91.64

noise and transformation, can be used with any font or style and does not need any segmentation. The proposed method is based on patch representation and non-negative matrix factorization and incorporates the global and local properties of images for accurate categorization. Although, the proposed method works for many types of layouts, it fails in document images where the text lines have different orientations. In the future we will investigate new ways to generalize the proposed method to more complex situation where different fonts or styles with different and more complex layouts in different scripts are used to create that document image.

Acknowledgment

The authors would like to thank NSERC of Canada for their financial support.

CHAPTER 5

ARTICLE III - PBLI: PATCH-BASED ARABIC SCRIPT LANGUAGE IDENTIFICATION USING NON-NEGATIVE MATRIX FACTORIZATION

Ehsan Arabnejad[1], Mohamed Cheriet[1]

[1] Synchromedia Laboratory, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

Submitted to Elsevier Pattern Recognition

Abstract

One of the main steps in automatic understanding of a document image is obtaining information about the language of that document image. This information is useful for document image categorization and also necessary for the Optical Character Recognition (OCR) engines. In this paper, we propose a novel method for language identification of printed documents and ancient manuscripts using Non-negative Matrix Factorization framework. The first step of the algorithm is extracting patches from the series of document image and second step is creating codebooks using these patches that is used for representation and classification. We used Non-negative Matrix Tri-Factorization for simultaneous feature extraction and clustering in which the algorithm is alternating between feature extraction and clustering to explore the interrelation between features and samples. The proposed method is robust against noise and degradation, extracts features and create codebooks simultaneously so the features are more robust, the codebooks are more representative and the representation is more discriminant. We tested our method on two datasets of digital-born documents dataset composed of 9 and ancient manuscripts dataset composed of 3 Arabic script based languages. Experiments conducted on these datasets show the robustness and the flexibility and comparison with the state-of-the-are methods demonstrate the accuracy of the proposed method.

Keywords

Language identification, Patch representation, Non-negative Matrix Factorization, Bi-clustering

5.1 Introduction

Language identification is one of the fundamental branches of document image processing field and one of the crucial steps in automatic document understanding. This system is necessary in many applications such as multilingual document image processing where the goal is automatic search, indexing and etc. Many Character Recognition Systems (OCR engines) are proposed and developed to handle the demand of the digital world for automatic understanding of a document image but most of the engines need information about the script and/or language of the document image. The previous research in this field is mostly focused on the Latin script based languages such as English, French, and etc. where the elements of the script (letters) are isolated and in their cursive style the characters can be tracked and separated in the word with a high accuracy.

The language identification is more challenging in the languages that the nature of their script is cursive. Arabic script is one of these examples that shared between many languages such as Arabic, Persian and etc. This script contains 28 letters and some of the letters are only different in the number and/or position of dots. In addition to that the shape of letter should also be changed based on the position of the letters in the word, that takes one of these forms such as individual, final, beginning and middle forms.

Segmentation of words into letters is a difficult task and many OCR engines are recently proposed to avoid the needs for segmentation. As theses engines are trained on the sequences of the characters, they also extract information surrounding the characters to accurately predict the current character. These sequences of the characters are language dependent and the accuracy of recognition depends on selecting the OCR engine trained on the samples of the same language. The information about the language of a document image can also be used as post-processing step i.e. a dictionary will be used (after the recognition step) to correct the errors of the recognition step. Performance of such post-processing tasks highly depends on the number of the errors in recognition step.

The best strategy for having a high quality recognition and avoiding these types of errors is using a bank of word-character recognition systems, where each recognition system is adapted to specific language, combined with a language identification system that selects a proper word/character recognition system based on the language of the document.

Language identification methods can be divided into two categories based on type of data:

- Text-based methods where the textual information of document image is available. In these methods, the n-gram representation or keyword representation followed by some other techniques -to reduce the influence of the error- is used for language identification.
- Image-based methods where the component of the document images are coded by some measures and few features are extracted based on this coding and the relation with textual information and these features are used for document image representation and language identification.

Text-based methods are based on the assumption that the textual information of the document is available, either the document is digital born or the textual information is obtained by OCR/recognition system. For the digital born documents the textual information are accurate and the n-gram features or the frequent words as histogram or textual data processed by more advanced method is used for classification. The OCR based methods are based on the assumption that the accuracy of the OCR engine is high or at least the error produced by OCR is consistent (not random error) so the proposed methods use the similar features or representations for language identification. One note for these type of methods is that all of the images should be processed by one OCR engine and this engine should be changed, altered or modified.

Image based methods -which are mostly developed for Latin-based languages- are based on the assumption that the component (individual letters) of a document image can be separated from the rest of the letters in a word. These components are codded (based on some observations) by defining some features such as ascenders, descenders and few other distinctive features for grouping the components into few categories. Then the sequence of letters (words) is codded by these categories and this coding is compared to the textual information to find the most frequent words and their corresponding codings. Then, the language of a document image can be identified by comparing this information to the set of labeled information.

Obtaining information about the components of a document image (letters) is a difficult task and involves many complex processes such as layout analysis, segmentation and etc. The current image based methods are mostly focused on Latin-based languages where the segmentation to individual letters is not very hard. For Arabic script based languages the components (letters) attach to each other and also change their shapes according to the position in the word. Detecting these objects needs intensive processing steps of layout analysis and segmentation in different levels which is a very complex task for such a cursive script. From another point of view, most of these segmentation methods are based on the assumption that the script/language of the target image is known.

Our approach is based on the fact that the differences between languages can be captured by tracking how the elements of the script/language are used to construct words and sentences (similar to second category). In order to avoid the complexity of the layout analysis, we propose to use patches as the unit of information. Extraction of these patches does not need any complex segmentation method and the patches contain information about the elements of the script/language (letters). If the patches processed in a right way, the information about the letter that covered by the patch and also information about the surrounding letters can be extracted and used for language identification.

From another point of view, the procedure of designing features for objects coding is as follows, an expert observes the objects (letters of the script) and defines some features that can be used for dividing these objects into few groups. The features are designed in the ideal case (no noise and degradation) so they cannot be measured accurately in the presence of noise. Any new language or new fonts/style needs opinion of expert for approving features or designing new features. Here we propose to use a method based on Non-negative Matrix Factorization (NMF) for extracting features and grouping (clustering) of objects to create codebooks. The proposed method can be used for any script and any type of fonts/style.

The organization of this paper is as follows. In section 5.2 related work and state of the art methods are discussed. Then the motivation and the proposed method are presented in section 5.3 and section 5.4. The experiments and evaluation are presented in section 5.5. Finally, the summary of the work is presented in the section 5.6.

5.2 Related Works

5.2.1 Bag-of-words model

In the bag-of-words model, a document is represented as a collection of words (order is not important) and the representation is the frequency of words in that document. The bag-of-word model is used with images and an image will be treated as a document. The above process includes the following steps:

- feature detection
- feature description
- codebook generation
- representation

The first step is feature detection or keypoint detection in which the goal is to find locations on the image that have specific properties and these properties do not change under some level of transformation. In feature description step a descriptor (transform invariant) will be extracted from those locations. The next steps for the BOW model are creating dictionary of codewords from this set of descriptors and representing images with those codewords.

5.2.2 n-gram

The n-gram feature is defined as continuous sequence of n objects from a string of text. The objects here can be letters, words or any pairs based on the application. The well-known first 3 level features are uni-gram, bigram and trigram. This model is used to create a probabilistic language model to predict the future letters or words based on a sequence of objects. These features are very simple and can be scaled by using larger n and used in several fields of computer science, such as information retrieval, cryptanalysis, language identification of text and etc.

5.2.3 clustering

One of the fundamental topics in data mining and statistical learning is clustering i.e. partitioning the available data into different groups without having any information about the groups (unsupervised). The conventional methods such as k-means are proposed for one side clustering, clustering of samples (most common) or features. In many real word applications the relation exists between samples and features and this relation can be exploited by clustering of samples and features simultaneously, for example in dataset of documents and words the goal is to cluster similar words and also at the same time similar documents. In this kind of situation the clustering task on data and features are co-related and traditional method hardly able to capture this relation and dependency between data and the features. The aim of co-clustering is to cluster both data and features at the same time by exploiting the interrelation between them. Advantages of co-clustering over the clustering is shown in many applications.

5.2.4 Language Identification in Literatures

In this section we present state-of-the-art methods of language identification and we discuss major problems and limitations of these methods. As it is mentioned before, language identification methods can be divided into two groups, text-based methods and image-based methods.

Text-based language identification: In the ideal case where the textual information of document image is available n-gram features are used for representation and language identification.

In Hakkinen & Jilei (2001a) a method for language identification based on n-gram and decision tree is proposed. Decision tree trained on the context around each character (previous character and next character) using set of lexicons tagged by language is used for classification. After the predicting the language label for each letter in a word level, the final decision for text line or a paragraph is based on majority voting.

In Kruengkrai *et al.* (2005) language identification based on string kernel is proposed. In Bilcu & Astola (2006) a method based on combination of MLP and decision rule is proposed to separate English words from French words. After analyzing 4 adjacent letters, the authors observed that 75% of these combinations are unique to specific language and they have used decision rules to classify words based on these rules and if the word cannot be classified in the first step MLP which is trained based 5 letters combination is used to classify that word.

In Hakkinen & Jilei (2001a) language identification method using decision tree and ARTMAP is proposed for Arabic script language identification. The first step is separating Arabic and Persian using the unique letters of Persian and decision tree combined with ART neural network for language identification. In Selamat & Ng Choon (2008) the previous work is improved by using fuzzy ART and also Urdu text is also added to the data.

In Kruengkrai *et al.* (2005) the authors proposed to use letter frequency document frequency features (LFDF) for finding the most appeared sequence of characters and then these weighting combined with the dataset from Wikipedia is used to separate Arabic script based languages.

In Takçı & Güngör (2012) a method for language identification using inverse class frequency is proposed for Latin-based language identification.

In Abainia *et al.* (2016) language identification method based on statistical approaches is proposed. The language identification in different levels such as letter and word level combined with language identification based on special letters for each language is used for identification. The authors used two hybrid algorithms with the mentioned features and n-gram features for language identification.

Image based language identification:

One of the pioneering methods for language identification document image is introduced in Spitz (1997b). The authors proposed a two-step method for script and language identification. The script of document is determined in the first step by some features and then in the second step they proposed two language identification methods for Han-based languages and Latin based languages. For Han-based languages they used optical density function for feature extraction and Linear Discriminant Analysis for classification. For Latin-based languages they used method based on finding characteristic word shapes coding by mapping characters based on their relative position to the baseline and x-line to a small number of distinct codes. Then the code words are used along with 15,000 words to find most frequent words and corresponding codes in each language. Then LDA is applied on the frequency features for language identification.

In Shijian & Chew Lim (2007) a method for language identification of printed document image is proposed. The authors introduced 7 categories of coding and words are coded based on their belonging to these categories and the document image is converted to a vector by this coding and their frequencies. The languages templates are created based on document vectors and cosine similarity is for classification.

In Farooq & Govindaraju (2007) the method for language identification of Arabic and Persian-Afghan document is proposed. This method is based on texture features and features based on Gabor transform in different scale and orientation combined with kernel-SVM classifier is used for separating Persian and Arabic blocks.

In Lu & Chew Lim (2008a) a two-step method for script and language detection is proposed. The first step is script identification and the second step is language identification of Latin based languages based on word shape features. This shape code is based on extremum points of the characters then -similar to method in Spitz (1997a)- code/frequency representation of document images is created based on shape codes and language template is used for classification.

The drawback of text-based method is that if the error of OCR engine is high the performance of language identification is low. Layout analysis has a huge effect on the performance of the OCR engine and these methods will be influenced by the error in layout analysis step. The text should be obtained by the same OCR engine for all of document images.

The limitation of current image-based methods are that method based on texture need intensive normalization and most of methods are proposed for printed Latin script where the characters are isolated and segmentation is not hard. For Arabic script in which the words are created by attaching characters many of these methods cannot be applied.

5.2.5 Non-Negative Matrix Factorization (N-NMF)

One common idea behind different approaches for noise removal is replacing the original data by a lower-dimensional representation using subspace approximation. Often the target data is non-negative and this non-negativity should be preserved in approximation. This problem is known as non-negative matrix factorization (NMF) and it is defined as follows: Given a non-negative matrix (data matrix) $X \in R^{m \times n}$ and a positive integer $k < min(m,n)$, the goal is to find non-negative matrices $W \in R^{m \times k}$ and $H \in R^{k \times n}$ such that:

$$X = WH \tag{5.1}$$

In many applications the clustering power of NMF is explored so it can be used for clustering of samples or clustering of features (depending on application). The main difference between the NMF and k-means is that the former is a soft clustering method which means sample can have relation with more than one clusters with a weight that shows the relation while k-means is hard clustering method and the output is cluster index.

A similarity measure is needed to quantify the quality of approximation. The primary and mainly used distance functions for NMF are Euclidean distance (EUD) Lee & Seung (1999) and Kullback–Leibler divergence (KLD) Lee & Seung (1999). Other distance functions such as Minkovsky family of metrics or l_p-norm, earth mover's distance Sandler & Lindenbaum (2011), $\alpha-$ divergence Cichocki *et al.* (2008), $\beta-$ divergence Kompass (2007), $\gamma-$ divergence Cichocki *et al.* (2006), Bergman distance Dhillon & Sra (2005), and $\alpha-\beta-$ divergence Cichocki *et al.* (2011) are used for specific applications.

NMF with different constraints are used in many applications, the constraints are sparsity constrains in Hoyer (2002), Orthogonality constrains of bases in Choi (2008), Discriminant constrains in Wang & Jia (2004), and manifold regularity constrains in Cai *et al.* (2011).

In other applications, the factorization is modified to adapt the NMF to the specific problem such as Weighed NMF in Kim & Choi (2009), Convolutive NMF in Smaragdis (2007) and Non-Negative Matrix tri-factorization in Yoo & Choi (2010).

The methods such Semi NMF in Ding *et al.* (2010), Non-Negative Tensor Factorization in Shashua & Hazan (2005), Non-Negative Matrix-set Factorization in Li & Zhang (2007) and Kernel NMF in Lin (2007) are proposed to generalize NMF to other types of data.

Original NMF has been solved by Multiplicative algorithm in Lee & Seung (2001) where two multiplicative update rules are used to minimize the objective function iteratively and alternatively:

$$H_{kj}^{(t+1)} = H_{kj}^{(t)} \frac{(W^T X)_{kj}}{(W^T W H^{(t)})_{kj}} \tag{5.2}$$

$$W_{ik}^{(t+1)} = W_{ik}^{(t)} \frac{(X H^T)_{ik}}{(W^{(t)} H^T H)_{ik}} \tag{5.3}$$

Where t is iteration index.

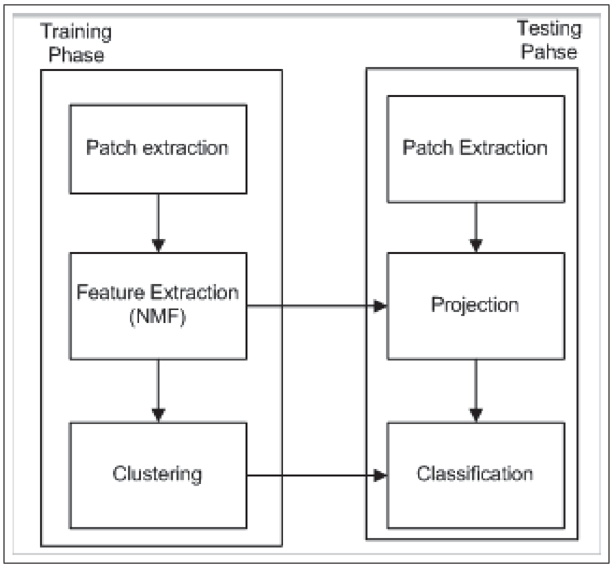


Figure 5.1 Flowchart of proposed method

In Ding *et al.* (2006) Non-negative Matrix Tri-Factorization is proposed by adding new variable to the factorization and it has been shown that the effect is bi-clustering i.e. simultaneous clustering of rows and columns of data matrix. In Yoo & Choi (2010) Orthogonal NMTF (ONMTF) method is proposed by imposing orthogonality constraints on the two of the three factors and the updating rule based on stifled manifold is introduced.

5.3 Motivation

The main difference between script identification and language identification is that different scripts have different components (letters) but different languages may use the same components (if they share the script). In script identification the difference can be obtained by features from the shape of components but in language identification the difference is in the sequence of them or how these elements are used to create words and sentences. So it is not possible to differentiate languages by just comparing features of the components.

The aim in language identification is assigning labels (codes) to all textual objects (letters) by OCR engines in text-based methods or by generating codebook in image-based methods and then analyzing the sequence of labels to differentiate languages. The labels are not important by themselves here and as long as they are consistent (not random) they can be used for language identification. The challenge of dealing with Arabic script is that segmentation of words into characters is very difficult and the coding strategy (proposed for Latin-based language identification) cannot directly be applied.

In Zhu *et al.* (2009) a method for language identification of handwritten documents image based on codebook is proposed with intuition that the segments obtained from the edges of the textual component are informative for representation of the document image. In their method languages are selected from different scripts so the main task is script identification. In this method layout analysis step is skipped but the features (3 adjacent segments) that are extracted are very local and do not provide enough information about the letters of the script.

In order to overcome the segmentation problem, we propose to use patches as the lowest level of information. Image patches are robust to discontinuity, noise and degradation and can be extracted by different methods. If the patches are selected properly (with proper location and the proper size) they cover a character or part of the word so they will provide a good representation for document image and this representation can be used for further processing such as language identification.

Manual designing of features is hard and time-consuming task the features should be revised if the script (languages), the writing style or dataset changes. Here we propose to use Non-negative Matrix Tri-Factorization. NMTF is selected for two purposes, clustering of features for feature extraction and clustering of samples for creating codebook. As it is mentioned before, co-clustering methods are able to find the interrelation between samples and the features.

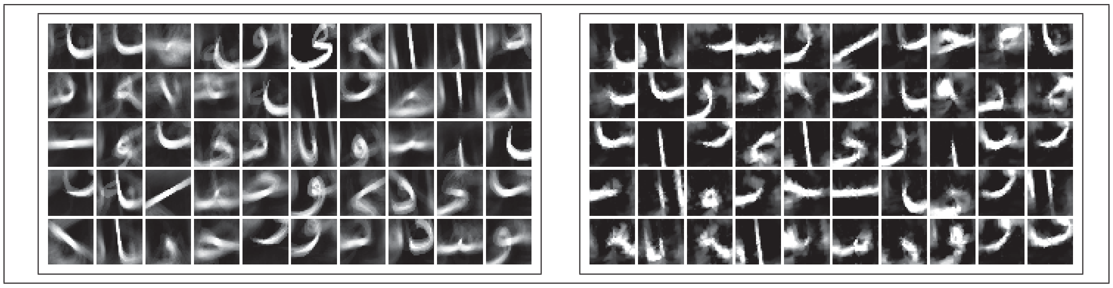


Figure 5.2 50 cluster centers obtained by Left) kmeans, right) NMTF with 100 bases

5.4 Proposed Method

The proposed method for language identification is illustrated in Figure 5.1. After pre-processing and preparing data, several patches are extracted from the image blocks and then NMTF is used to learn bases and clusters from this set of patches. For the test set, the similar process is done for extracting patches, and the bases that are learned in the training step are used to extract features. For each document image a histogram is created based on the coefficients obtained by NMTF. Then K-nearest neighbor algorithm (K-NN) is used to classify new features based on proximity to the nearest feature point in the training set.

5.4.1 Preprocessing

To prepare data for language identification, the image blocks (color) are converted to gray level images and then converted to binary image (black and white) image using Otsu Otsu (1979) method. These image blocks are skew corrected by analyzing the variance of projection of the pixels in different angles.

5.4.2 Patch extraction

One of the main steps in bag-of-word model is extracting keypoints. Some methods such as SURF Bay *et al.* (2008) are looking for the keypoints based on corner detection. In this paper we tested 3 methods for patch extraction, Skeleton Patches and SURF patches. The skeleton patch is proposed in Farrahi Moghaddam *et al.* (2012) and uses the skeleton map of the foreground pixels as center points for the patches.

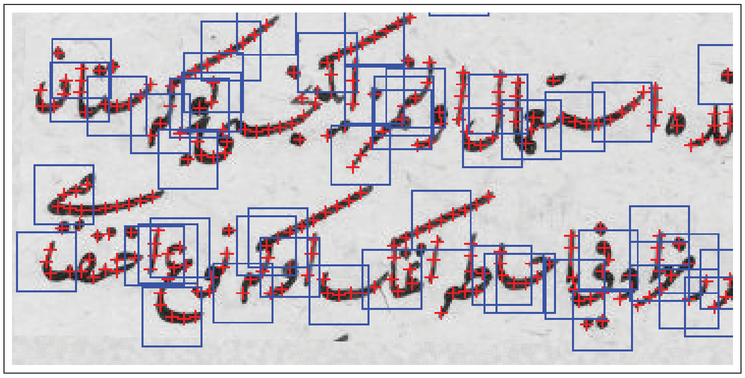


Figure 5.3 Sample image and keypoints extracted from the skeleton map

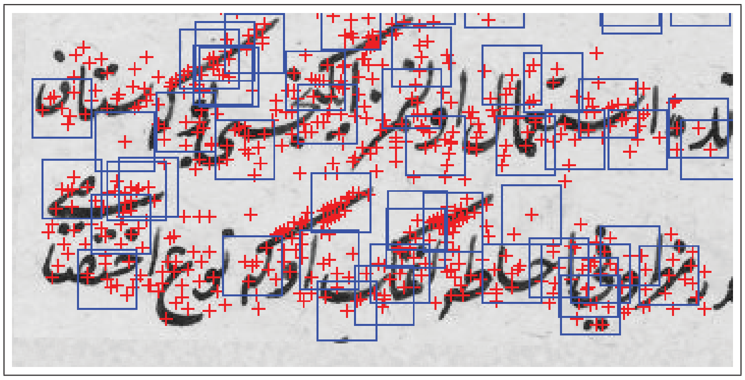


Figure 5.4 Sample image and keypoints extracted by SURF method

5.4.3 Feature Extraction - dimensionality reduction

One strategy for language identification is bag-of-visual words, where some components of the image are extracted and clustered to find a set of components called codebook (dictionary). These components will be used to represent an image by a histogram that shows the presence/frequency of those components in it. One of the main features of NMF is soft clustering which means the data points can be related to more than one cluster. Soft clustering methods show their potential and performance in many applications such as text clustering. Other main features of NMF is part based reconstruction that obtained by non-negativity constrain of objective function. This feature is well suited for representation of an object in an additive manner.

Based on our knowledge there is no method for language identification of ancient manuscripts based on NMF. Here we propose to use Non-Negative Matrix Tri-Factorization (NMTF) proposed in Yoo & Choi (2010) based on the idea that the NMTF approximation is obtained by clustering of rows and columns of data matrix X with matrices F, S and G.

$$X = FSG^T \tag{5.4}$$

The objective function for NMTF is:

$$\min_{F>=0,S>=0,G>=0} \frac{1}{2} \left\| X - FSG^T \right\|_2^F \tag{5.5}$$

and the multiplicative update rules for three matrices introduced in Yoo & Choi (2010) are as follows:

$$F = F \odot \frac{\left(XGS^T\right)}{\left(FSG^T X^T F\right)} \tag{5.6}$$

$$S = S \odot \frac{\left(F^T XG\right)}{\left(F^T FSG^T G\right)} \tag{5.7}$$

$$G = G \odot \frac{\left(X^T F S\right)}{\left(G S^T F^T X G\right)} \tag{5.8}$$

Orthogonality Constraint on the G is forcing the algorithm to learn local and independent bases. Orthogonality constraint on the F is forcing this matrix to be sparse and in the ideal case cluster indicator. For the simple patches the F matrix shows only one or two non-zero elements but for the complex patches that are combination of more parts from two or more different letters, F has more non-zero elements and its coefficients show how the components in S are used to represent that patch.

By projection of patches to a lower-dimensional space, we obtain a new representation which is abstract and invariant (up to some level) to some transforms such as translation or rotation. This is very important in our case where the patches are extracted from the skeleton of the image and the difference of many patches is just a slight shift in the center point. In conventional clustering techniques the clustering is done in the original space so the translation effect is shown in the learned cluster centers. By using NMTF, data points are projected into lower-dimensional space and then clustered so by alternating between learning bases and learning clusters the learned bases provides better and abstract representation and the clusters centers are more informative.

The cluster centers obtained by K-means and NMTF methods are shown in Figure 5.5. We can see that some of the pixels for K-means cluster centers are blurry due to the fact that the shift in the center of the similar neighbor patches creates almost different patch from the K-means point of view. We can see that the NMTF cluster centers are visually better than Kmeans.

5.5 Experimental Results

The proposed language identification method is evaluated on two datasets. Different experiments are performed to show the performance and robustness of proposed method.

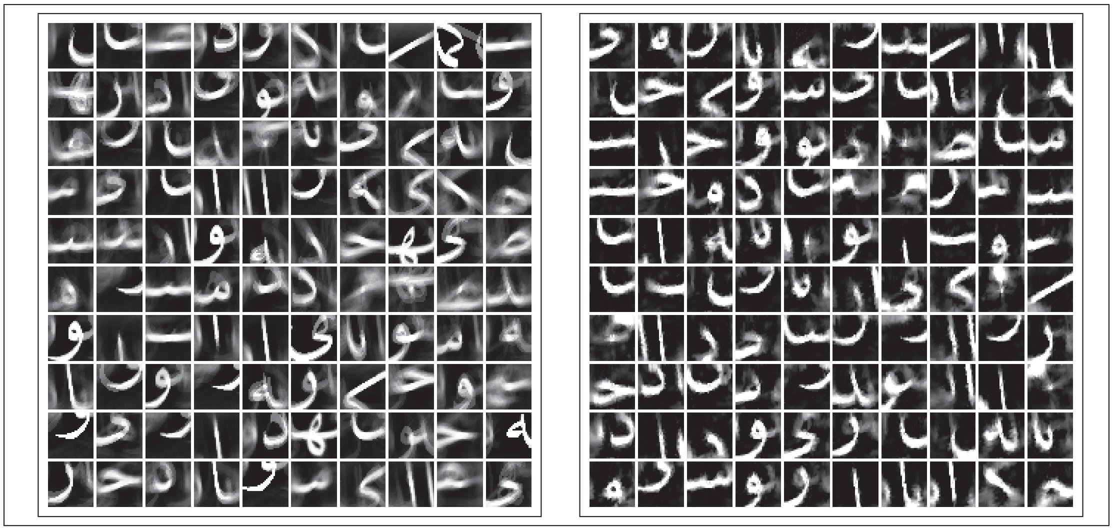


Figure 5.5 Cluster centers obtained by (left) Kmeans , (right) NMTF

5.5.1 Database

To evaluate the proposed method, the multi-language database has been created using document images of 2 different languages of Arabic script. This dataset consists of Arabic and Farsi languages. These document images are obtained from different digital libraries and belong to different times in history. Our aim is developing a database with different languages and also different writing style. Each document image in this database has been labeled based on language. Approximately 650 document images are selected for this experiment (train and test).

The second dataset is created by getting the Wikipedia pages (text) of 9 Arabic script languages i.e. Arabic, Persian, and etc. These pages are cleaned by removing all of non-Arabic characters and numbers. Sentences with the specific lengths of characters are randomly selected from these pages. Then the textual data are convert to images with the software that resembles some of the Arabic writing styles such as Naskh, Nastaliq and etc. The purpose of this dataset is to compare the proposed method with the text-based methods and n-grams features. The tables and figures are shown based on the following labels:

Table 5.1 languages and labels used in first set of experiments

Laguage	Label	Language	Label	Language	Label
Arabic	ar	Egyptian Arabic	arz	Azerbaijani(turki)	azb
Farsi	fa	Gilaki	gl	Mazandarani	mz
Pashtu	ps	Sindhi	sd	Urdu	ur

Gilaki and Mazandarani are very similar to Farsi and Arabic and Egyptian Arabic are very similar so the second version of this dataset is created by changing the labels of Egyptian Arabic to Arabic and the labels of Gilaki and Mazandarani to Farsi. This dataset is based on 6 labels and the results are shown according to the following labels:

Table 5.2 languages and labels used in second set of experiments

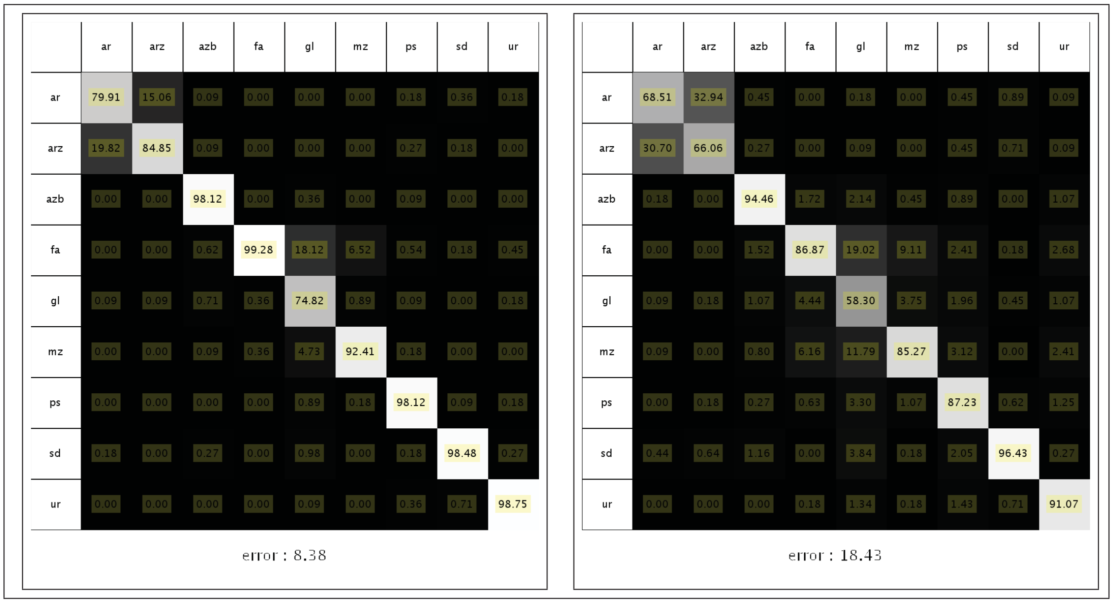
Laguage	Label	Language	Label	Language	Label
Arabic	ar	Egyptian Arabic	ar	Azerbaijani(turki)	azb
Farsi	fa	Gilaki	fa	Mazandarani	fa
Pashtu	ps	Sindhi	sd	Urdu	ur

5.5.2 Experimental Results

Synthetic dataset

In order to test the n-gram features on this database, strings with different lengths (200 and 500) are selected randomly and tested with unigram and bigram features and the results are shown in Figures 5.6 and 5.7. We can see that with the lower length of the string the difference between classification performance of using unigram ad bigram features is high but when we use higher length the classification is more accurate with both features and the accuracy difference becomes less. We can see from Figures 5.8 and 5.9 that when we change the labels from 9 to 6 the performance is better and this because of the similarity of mentioned languages.

In order to compare the proposed method with textual features, we create a dataset with the string length of 300 characters to evaluate text-based methods and image-based methods. The



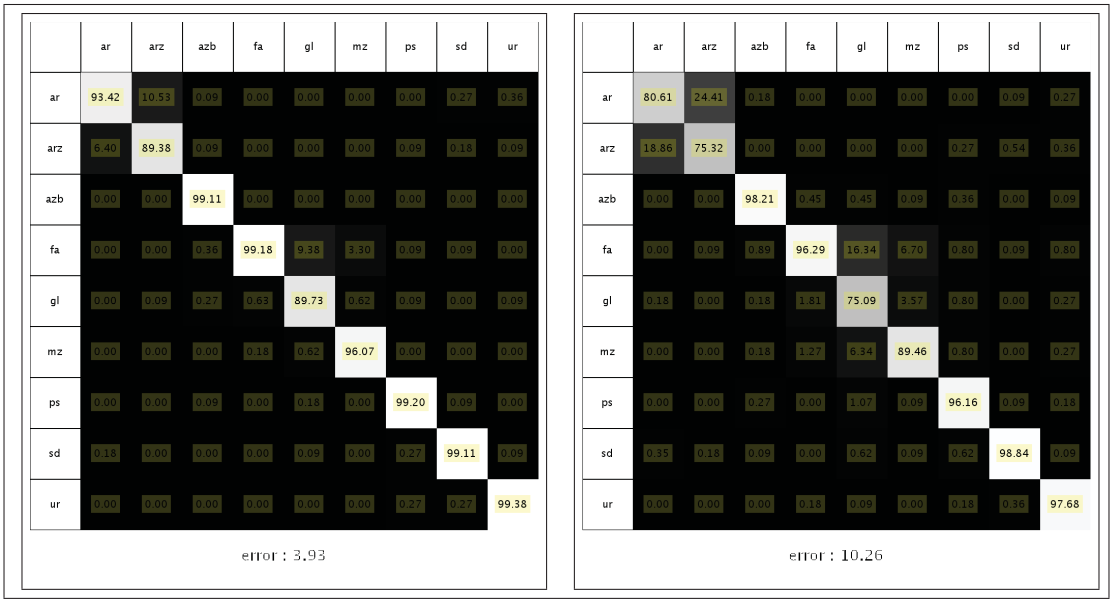
	ar	arz	azb	fa	gl	mz	ps	sd	ur
ar	79.91	15.06	0.09	0.00	0.00	0.00	0.18	0.36	0.18
arz	19.82	84.85	0.09	0.00	0.00	0.00	0.27	0.18	0.00
azb	0.00	0.00	98.12	0.00	0.36	0.00	0.09	0.00	0.00
fa	0.00	0.00	0.62	99.28	18.12	6.52	0.54	0.18	0.45
gl	0.09	0.09	0.71	0.36	74.82	0.89	0.09	0.00	0.18
mz	0.00	0.00	0.09	0.36	4.73	92.41	0.18	0.00	0.00
ps	0.00	0.00	0.00	0.00	0.89	0.18	98.12	0.09	0.18
sd	0.18	0.00	0.27	0.00	0.98	0.00	0.18	98.48	0.27
ur	0.00	0.00	0.00	0.00	0.09	0.00	0.36	0.71	98.75

error : 8.38

	ar	arz	azb	fa	gl	mz	ps	sd	ur
ar	68.51	32.94	0.45	0.00	0.18	0.00	0.45	0.89	0.09
arz	30.70	66.06	0.27	0.00	0.09	0.00	0.45	0.71	0.09
azb	0.18	0.00	94.46	1.72	2.14	0.45	0.89	0.00	1.07
fa	0.00	0.00	1.52	86.87	19.02	9.11	2.41	0.18	2.68
gl	0.09	0.18	1.07	4.44	58.30	3.75	1.96	0.45	1.07
mz	0.09	0.00	0.80	6.16	11.79	85.27	3.12	0.00	2.41
ps	0.00	0.18	0.27	0.63	3.30	1.07	87.23	0.62	1.25
sd	0.44	0.64	1.16	0.00	3.84	0.18	2.05	96.43	0.27
ur	0.00	0.00	0.00	0.18	1.34	0.18	1.43	0.71	91.07

error : 18.43

Figure 5.6 Language Identification Accuracy using Bigram and Unigram representation on the sentence length of 200 characters



	ar	arz	azb	fa	gl	mz	ps	sd	ur
ar	93.42	10.53	0.09	0.00	0.00	0.00	0.00	0.27	0.36
arz	6.40	89.38	0.09	0.00	0.00	0.00	0.09	0.18	0.09
azb	0.00	0.00	99.11	0.00	0.00	0.00	0.00	0.00	0.00
fa	0.00	0.00	0.36	99.18	9.38	3.30	0.09	0.09	0.00
gl	0.00	0.09	0.27	0.63	89.73	0.62	0.09	0.00	0.09
mz	0.00	0.00	0.00	0.18	0.62	96.07	0.00	0.00	0.00
ps	0.00	0.00	0.09	0.00	0.18	0.00	99.20	0.09	0.00
sd	0.18	0.00	0.00	0.00	0.09	0.00	0.27	99.11	0.09
ur	0.00	0.00	0.00	0.00	0.00	0.00	0.27	0.27	99.38

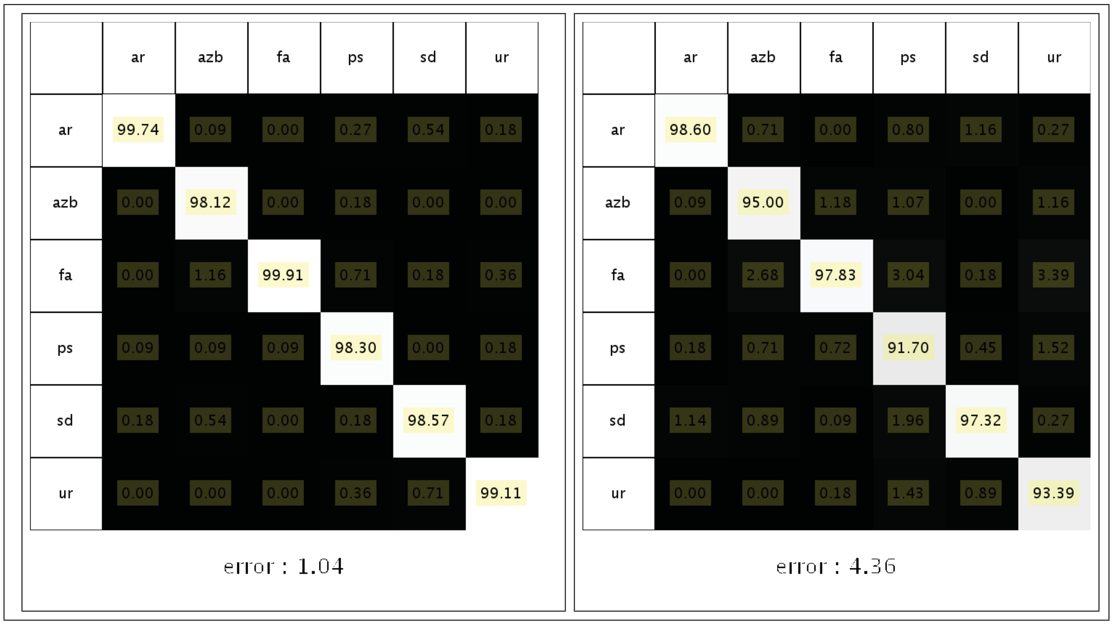
error : 3.93

	ar	arz	azb	fa	gl	mz	ps	sd	ur
ar	80.61	24.41	0.18	0.00	0.00	0.00	0.00	0.09	0.27
arz	18.86	75.32	0.00	0.00	0.00	0.00	0.27	0.54	0.36
azb	0.00	0.00	98.21	0.45	0.45	0.09	0.36	0.00	0.09
fa	0.00	0.09	0.89	96.29	16.34	6.70	0.80	0.09	0.80
gl	0.18	0.00	0.18	1.81	75.09	3.57	0.80	0.00	0.27
mz	0.00	0.00	0.18	1.27	6.34	89.46	0.80	0.00	0.27
ps	0.00	0.00	0.27	0.00	1.07	0.09	96.16	0.09	0.18
sd	0.35	0.18	0.09	0.00	0.62	0.09	0.62	98.84	0.09
ur	0.00	0.00	0.00	0.18	0.09	0.00	0.18	0.36	97.68

error : 10.26

Figure 5.7 Language Identification Accuracy using Bigram and Unigram representation on the sentence length of 500 characters

images of this dataset are processed and classified by 3 methods for creating codebooks i.e. kmeans, random patches and NMTF and the results for Bigram, kmeans and NMTF are shown



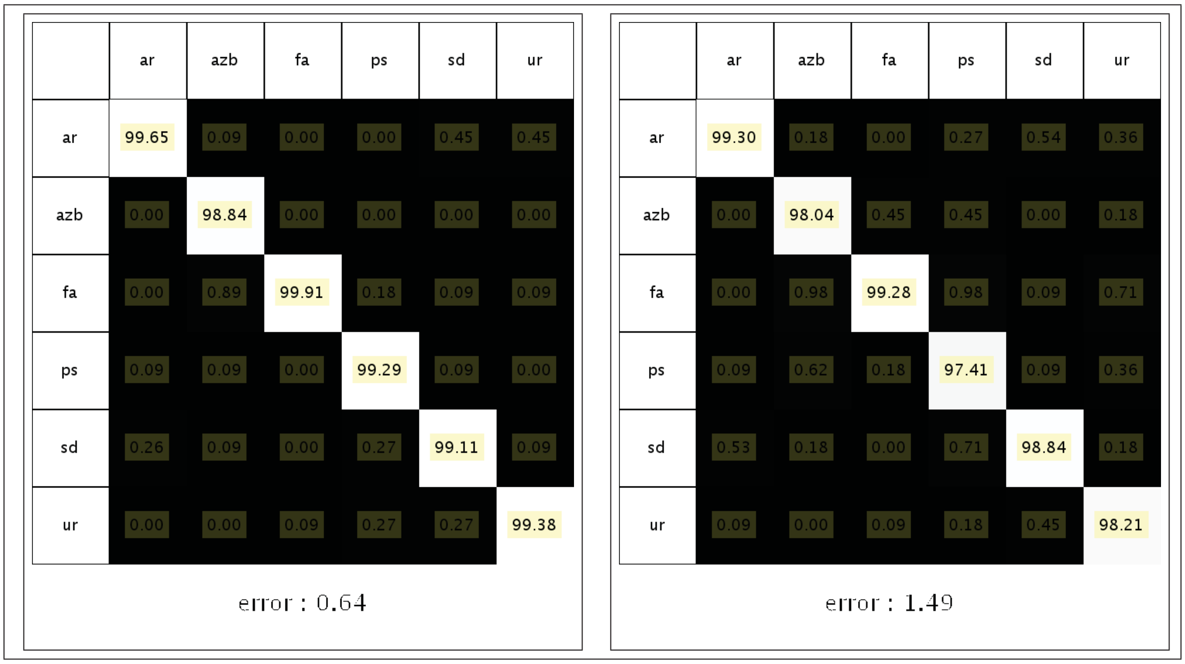
	ar	azb	fa	ps	sd	ur
ar	99.74	0.09	0.00	0.27	0.54	0.18
azb	0.00	98.12	0.00	0.18	0.00	0.00
fa	0.00	1.16	99.91	0.71	0.18	0.36
ps	0.09	0.09	0.09	98.30	0.00	0.18
sd	0.18	0.54	0.00	0.18	98.57	0.18
ur	0.00	0.00	0.00	0.36	0.71	99.11

error : 1.04

	ar	azb	fa	ps	sd	ur
ar	98.60	0.71	0.00	0.80	1.16	0.27
azb	0.09	95.00	1.18	1.07	0.00	1.16
fa	0.00	2.68	97.83	3.04	0.18	3.39
ps	0.18	0.71	0.72	91.70	0.45	1.52
sd	1.14	0.89	0.09	1.96	97.32	0.27
ur	0.00	0.00	0.18	1.43	0.89	93.39

error : 4.36

Figure 5.8 Language Identification Accuracy using Bigram and Unigram representation on the sentence length of 200 characters (6 langauges)



	ar	azb	fa	ps	sd	ur
ar	99.65	0.09	0.00	0.00	0.45	0.45
azb	0.00	98.84	0.00	0.00	0.00	0.00
fa	0.00	0.89	99.91	0.18	0.09	0.09
ps	0.09	0.09	0.00	99.29	0.09	0.00
sd	0.26	0.09	0.00	0.27	99.11	0.09
ur	0.00	0.00	0.09	0.27	0.27	99.38

error : 0.64

	ar	azb	fa	ps	sd	ur
ar	99.30	0.18	0.00	0.27	0.54	0.36
azb	0.00	98.04	0.45	0.45	0.00	0.18
fa	0.00	0.98	99.28	0.98	0.09	0.71
ps	0.09	0.62	0.18	97.41	0.09	0.36
sd	0.53	0.18	0.00	0.71	98.84	0.18
ur	0.09	0.00	0.09	0.18	0.45	98.21

error : 1.49

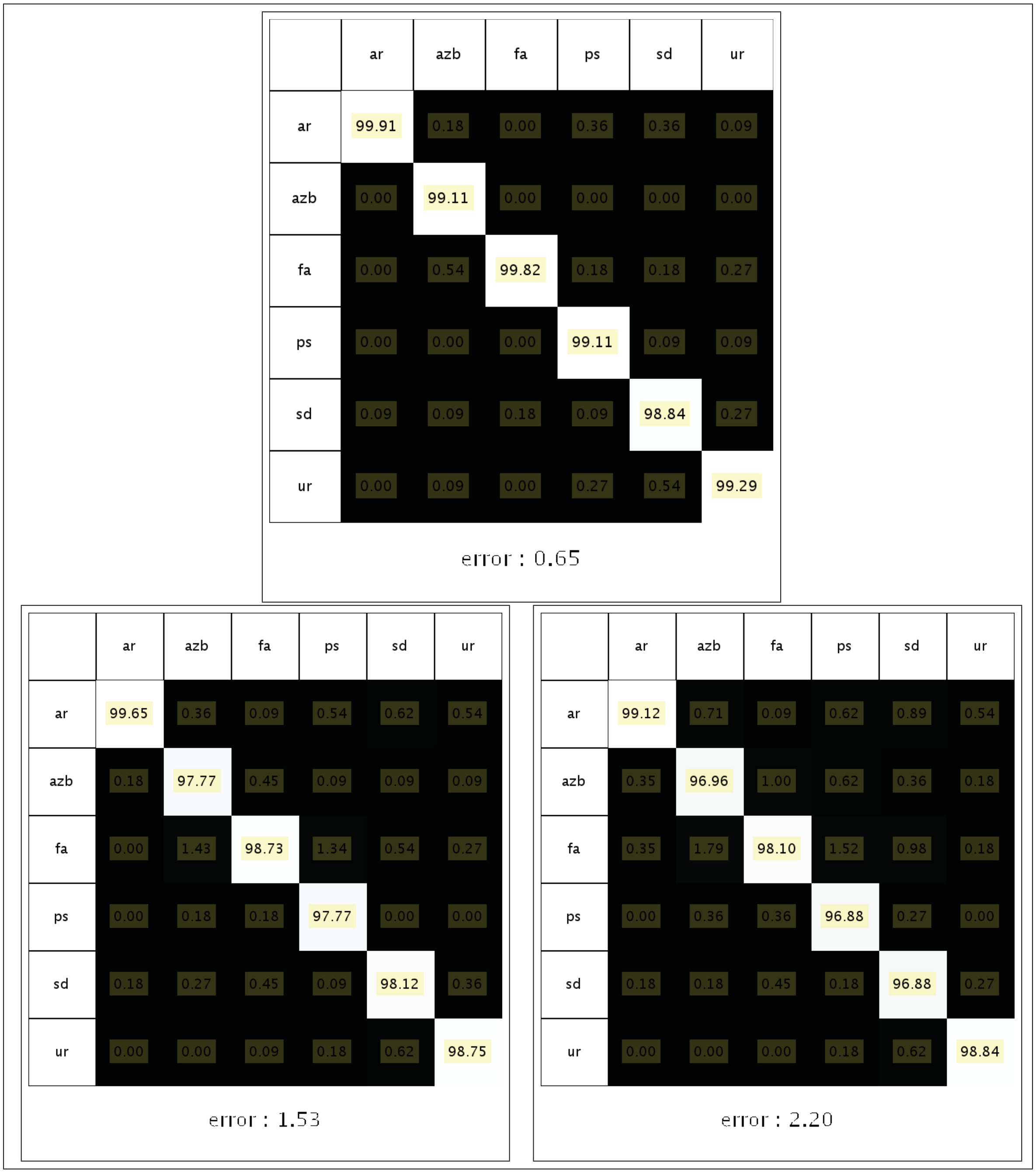
Figure 5.9 Language Identification performance using Bigram and Unigram representation on the sentence length of 500 characters(6 langauges)

in Figure 5.10. In order to have comparable result between text-based features and the image-based features dots are removed from the text which means the characters that have same shape but different dots are replaced with the one without dots. Also some letters are only different in individual or final form so this replacement is also done for this dataset. The n-gram based features obtained for this dataset depend not only on the characters but also the different forms of the characters. We can see that for the similar languages (or the languages that have many common words) the classification error is higher compared to the others. The patch based representation is obtained by NMTF $k = 1000$ and $l = 200$ and Kmeans with $k = 1000$. We can see that the bigram based features has the best results and NMTF is second with a very small difference.

In order test the sensitivity of the proposed method to languages that the the font does not exists in the training data, we performed an experiment as follows: for synthetic dataset (6 languages and 3 different fonts we randomly keep the data of document images of languages with one font and skip the other fonts, forexample for Arabic set we only kept the data related to Naskh font and for Farsi we kept the data of Nastaliq font and etc. Then, we created a mapping form the image data to textual data i.e. we consider:

$$H = PW \tag{5.9}$$

Where H is concatenation of n-gram histograms of images, P is histogram obtained by NMTF and W is projection matrix that obtained from training data by using $W = P^{-1}H$. The classifier is trained on n-gram features and the for each test image the corresponding patch representation is used with W to obtain H_t. The results of this experiment are shown in figure 5.11. We can see a decline in the results compared to data of patch representation but this results obtained by using the small portion of data. Analyzing the results showed that most of the errors are related to the images that contain names or words instead of sentences.



	ar	azb	fa	ps	sd	ur
ar	99.91	0.18	0.00	0.36	0.36	0.09
azb	0.00	99.11	0.00	0.00	0.00	0.00
fa	0.00	0.54	99.82	0.18	0.18	0.27
ps	0.00	0.00	0.00	99.11	0.09	0.09
sd	0.09	0.09	0.18	0.09	98.84	0.27
ur	0.00	0.09	0.00	0.27	0.54	99.29

error : 0.65

	ar	azb	fa	ps	sd	ur
ar	99.65	0.36	0.09	0.54	0.62	0.54
azb	0.18	97.77	0.45	0.09	0.09	0.09
fa	0.00	1.43	98.73	1.34	0.54	0.27
ps	0.00	0.18	0.18	97.77	0.00	0.00
sd	0.18	0.27	0.45	0.09	98.12	0.36
ur	0.00	0.00	0.09	0.18	0.62	98.75

error : 1.53

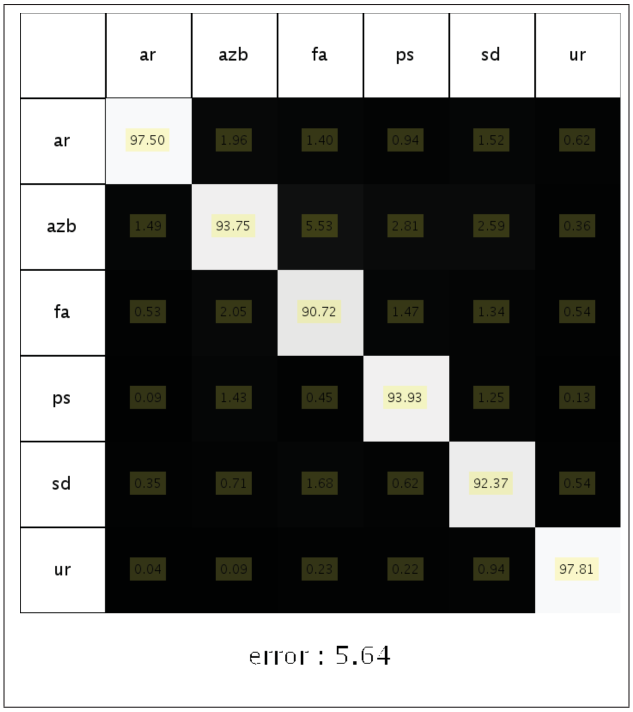
	ar	azb	fa	ps	sd	ur
ar	99.12	0.71	0.09	0.62	0.89	0.54
azb	0.35	96.96	1.00	0.62	0.36	0.18
fa	0.35	1.79	98.10	1.52	0.98	0.18
ps	0.00	0.36	0.36	96.88	0.27	0.00
sd	0.18	0.18	0.45	0.18	96.88	0.27
ur	0.00	0.00	0.00	0.18	0.62	98.84

error : 2.20

Figure 5.10 Classification result using Bigram feature for text-based method and NMTF (proposed method) and kmeans for image-based methods(sentence length: 300 characters)

Ancient manuscripts

The dataset which is created by 2 languages of Arabic and Farsi is tested with different methods and the results are shown in 5.10. We tested different similarity measures for comparing his-



	ar	azb	fa	ps	sd	ur
ar	97.50	1.96	1.40	0.94	1.52	0.62
azb	1.49	93.75	5.53	2.81	2.59	0.36
fa	0.53	2.05	90.72	1.47	1.34	0.54
ps	0.09	1.43	0.45	93.93	1.25	0.13
sd	0.35	0.71	1.68	0.62	92.37	0.54
ur	0.04	0.09	0.23	0.22	0.94	97.81

error : 5.64

Figure 5.11 Classification result with mapping from patch representation to n-gram representation (sentence length: 300 characters)

togram and also different number of nearest neighbor value. We observed that the Manhattan distance or city block with the value of 5 nearest neighbor produced the best result compared to Euclidean and correlation distance.

The drawback of using NMTF is that the iterative process should be used in order to find representation for new samples and this is a very time-consuming process. To speed up the algorithm we used the relation between clustering and NMTF so the initialization of F is done by using a procedure like k-means where the F is the matrix that only 1 element is non-zero and this is the cluster index. But if one element becomes zero in multiplicative update rule, it remains zero in NMTF steps so we used a very small value of e for the rest. With e equals to 0.001 we speed up the algorithm two times and that means the same representation is obtained with fewer iteration compared to random initialization.

As we discussed in previous section selecting proper values for k and l is very difficult and mostly data dependent. Here, we tested different number of bases k and different numbers for clusters l for NMTF algorithm. Based on experiment and the results that are shown in table 5.4 the best classification rate is obtained using 300 bases and 1500 clusters.

To show the performance of this method against other methods, the method based on texture feature, method based on the shape codebook and a methods based on n-gram features are implemented and used for language identification. Table 5.4 show the classification rate and comparison between algorithms and we can see that the proposed method outperformed the other two methods. Texture based method extracts global properties of images so the features also depends on layout and writing style. These methods are also sensitive to pre-processing steps. Method based on code book is very effective for script identification but as the segments are very local the representation is not discriminative for language identification.

Table 5.3 Performance of different algorithms on synthetic dataset (sentence length : 300 characterts)

Algorithm	Accuracy
uni-gram (text-based)	0.97
bi-gram (text-based)	0.99
Texture Method (Busch *et al.*, 2005)	0.75
shape codebook (Zhu *et al.*, 2009)	0.86
Proposed Method (patch-based)	0.98

Table 5.4 Performance of different algorithms on ancient manuscript

Algorithm	Accuracy
Texture Method (Busch *et al.*, 2005)	0.71
shape codebook (Zhu *et al.*, 2009)	0.82
Proposed Method	0.92

5.6 Conclusion

In this paper a novel method for language identification of printed/ancient manuscript based on image patches is proposed. This method does not need intensive normalization and is very robust against noise and degradation. The proposed method is a link between text-based method and image-based method which does not need any layout analysis and segmentation. The proposed method is flexible for different levels of identification and the experiments show the robustness and reliability of this method. In the future work we will investigate ways of using this approach for summarization of document image.

Acknowledgment

The authors would like to thank NSERC of Canada for their financial support.

CHAPTER 6

GENERAL DISCUSSION

This thesis has addressed the problem of categorization of document image and proposed novel methods for identification of different categories and different levels of layout with effective representation and feature extraction methodology. The literature review in Chapter 1 showed the different assumption and corresponding approaches for representation and feature extraction that used for script, font (style) and language identifications and it highlighted the limitations of current methods. Three questions were specifically investigated: a) What representation can be used to overcome the limitation of current method?, b) how to improve the description of the document image for categorization? And c) is there a general framework that can be used to represent and extract features form document images which can be generalized to different problems. The general methodology described in Chapter 2 highlights three research objectives that led to the development of 3 methods for script, style and language identification and also a framework for effective representation and feature extraction. First, a new method for script identification of ancient manuscripts based on patch representation is proposed and evaluated. Second, a new approach for font and style identification of printed and handwriting document image written in Arabic script is developed. Third, a new method for language identification of Arabic script based languages is proposed. These methods, the corresponding contributions and evaluations are discussed in Chapter 3, Chapter 4 and Chapter 5. In the following section, the advances that made in the state of the art for categorization, advantages, strength and limitation are discussed.

6.1 Script identification

Many methods are proposed and developed for script identification of document image based on different framework. Most of these methods are solving the problem in machine-printed documents and few of them addressed this problem in handwritten documents. All of these methods are based on the assumption that the level of the layout is known. This assumption imposes many limitations on practical usage of these methods. The second drawback of the current methods is that the features are designed by humans (expert) through observation which might not be generalizable. In our work, document images are represented with patches that reveal the local information of image and components of the script. Unlike all of the methods, our approach is not limited to work on specific level of layout. The patch-based representation provides a flexibility and robustness, flexibility of using with any level of layout and robustness against degradation and noise. This representation has the advantage that any error in the consequent steps can be analyzed and back tracked in order to improve the performance of identification (unlike some framework such as texture-based representation). One critical part of this approach is estimating the patch size. This parameter has an important role in accurate representation. Although, the feature extraction method used after patch representation is robust to some degree of scale and rotation transform, the patch representation used in this work is still in early stage and needs more improvements. The patch size is estimated from the global properties of the image therefore if there are irregularities in the layout the current strategy for size estimation might not return accurate size. Another assumption is that the document image is composed of horizontal textlines therefor we use skew estimation and correction method as a pre-processing step. Some scripts like Chinese and Japanese are composed of vertical textlines so the proposed method should be used carefully. For the curved textline this patch size estimation is not accurate. Another limitation of the proposed method is that amount of data that should be at least one textline. The reason is that individual word written in Latin script might be composed of only lower case letters and also without the characters that have some part under the baseline therefor the text height estimation is not accurate. Similar situation exists for Arabic script and some letters are written under the baseline and some letters have vertical stroke.

6.2 Font and style identification

In Chapter 4, the aim was to introduce an effective representation and feature extraction scheme for font and style identification in machine-printed and handwritten manuscripts. The current methods are following two main approaches, global-based features and local-based features. Global-based features are mostly inspired from texture approaches and define the document image as texture and then texture representation and feature extraction are adapted to this problem. Local-based approaches are mostly applied on components of document image and need segmentation of objects. The global approaches need many pre-processing steps and the local approaches need very accurate segmentation which is not easy for some fonts or style. The fixed level of layout is another major problem for many methods. We used a patch-based representation and NMF for feature extraction. The proposed method does not need intensive normalization (pre-processing) and patch size can be estimated from the global properties of document image that can be estimated with high accuracy. One of the critical steps of the method is setting patch size for representation. In the case of very irregular and complex layout, this method can not estimate accurate patch size. Another drawback is the classification procedure which will be computationally expensive if many classes are considered.

6.3 Language identification

In Chapter 5 that covered the third objective, we proposed a novel method for language identification of Arabic script languages. The state of the art methods for language identification fall into two groups, text-based approaches and image-based approaches. The general framework is BOW for these methods and the ultimate goal is to have a labeling procedure and use it to represent document (document image) with a histogram that shows the frequency of the components. The text-based approaches need many complex steps toward recognition and current image based approaches are applicable to scripts that are composed individual letters. Here we proposed new approach for language identification of Arabic script languages using patch-based representation and NMTF. Patch representation gives us the elements of script or combination of them and NMTF provides a robust and flexible framework for learning codebooks and extract features. The limitation of proposed method is that it needs retraining the algorithm to obtain the bases and codebooks if new font or style is added to the dataset.

6.4 Limitation

The general limitation of the proposed methods is that they are based on the assumption that the document image is composed of one category and in mixed situation, document images with more than one scripts and different fonts, it is not applicable. Another limitation is that if the text-height changes more than 10% of the average text height (font with different sizes) this method might miss classify that sample. These methods work with left to right or right to left script but cannot be used for top-down layout like Chinese. In the case of mixed layout or different direction of textline this method is not applicable.

6.5 Future Work

These proposed method together provide a framework for representation and feature extraction toward document image categorization. As mentioned before, one of the critical step of the proposed method is estimating text-height from global properties of image. In the future we will investigate the ways to generalize our method to be used on multi-category document image where different script and different fonts or styles is used in one page. This will be investigated by algorithm to estimate the patch size locally by measuring information at different patch size based on feature selection methods. The methods proposed for font and language identification are tested on Arabic script fonts and languages. In the future we will test our methods on other scripts. One of the ultimate goal of this paper is to investigate the the approaches for content-based document image categorization. The current methods are based on textual data that obtained using OCR systems. In the future we will investigate this framework for more complex categorization problem such as image abstraction or topic extraction based on patch representation.

CONCLUSION AND RECOMMENDATIONS

In this thesis, we addressed the problem of document image categorization, proper representation and feature extraction for the purpose of script, style and language identification in machine-printed and handwritten manuscripts. Proper representation and features are two critical parts of every pattern recognition system. These two aspects toward document image categorization have been studied in this thesis. We have introduced three methods for script, style(font) and language identification based on patch representation and NMF. These three methods are introduced in a sequence that have priority in recognition system. First, the global properties of the components (patches) determine type of the script and this can be achieved with learning features from the components. Then, the fine details of the components determine type of font or style and this difference can be captured by bases and clusters obtained from the specific class of font or style. Finally, the frequency of usage of these objects determine the corresponding language and this frequency is captured with an efficient and robust codebook and dictionary learning. For document image categorization, this thesis opens a path to a novel representation and feature extraction of document image. The current state of the art methods mostly follow the global based approaches for script and style identification and text based approaches for language identification. Although, the global based approaches provide convincing results, they impose many limitations due to fixed level of layout. Text based approaches for language identification are also relying on the performance of the recognition system that highly depends on many factors. Therefore, we proposed a framework for categorization of ancient and machine-printed manuscripts based on patch representation and automatic feature learning. The extracted patches cover partially or fully the building blocks of the scripts i.e. letters or symbols so the information related to the specific task of categorization can be extracted in a hierarchical way and automatically form these patches.

Summary of contribution

In this section, we briefly highlight the major contribution of this thesis.

- A new patch-based method for script identification of ancient manuscripts is proposed. This is the first approach that considers the patch based representation for the problem of script identification and provides flexibility and robustness for representation along with non-negative matrix factorization a base for automatic feature learning and extraction.
- A new patch-based method for font and style identification of printed and handwritten Arabic manuscript. This is the first approach that uses image patches as the lowest unit of information for the task of font and style identification. Automatic feature learning based non-negative matrix tri-factorization combined with patch representation provide the framework for robust, accurate and generalizable framework for the task of identification in different type of document images.
- A new approach for language identification of handwritten and machine-printed document image. To the best of our knowledge this the first method that introduced patch-based representation and new codebook generation and representation based non-negative matrix tri-factorization. This framework is flexible and robust and can be generalized to more complex task of categorization.

Articles in peer reviewed journals

1. Ehsan Arabnejad, Reza Farrahi Moghaddam, Mohamed Cheriet, PSI: Patch-based script identification using non-negative matrix factorization, In Pattern Recognition, Volume 67, 2017, Pages 328-339, ISSN 0031-3203
2. Ehsan Arabnejad, Mohamed Cheriet: PFSI: Patch-based Arabic script font and style identification using non-negative matrix factorization. Submitted to Pattern Recognition (2017).

3. Ehsan Arabnejad, Mohamed Cheriet: PBLI: Patch-based Arabic script language identification using non-negative matrix factorization. Submitted to Pattern Recognition (2017)

Book chapters

- Mohamed Cheriet; Reza Farrahi Moghaddam; Ehsan Arabnejad; Guoqiang Zhong: Manifold Learning for the Shape-Based Recognition of Historical Arabic Documents, Handbook of Statistics ISSN: 0169-7161, Volume 31 (2013)

Conference organization

- IEEE 11th International Conference on Information Sciences, Signal Processing and their Applications (ISSPA '12), volunteer.

Paper reviewing

- 9th International Workshop on Systems, Signal Processing and their Applications (WOSSPA '13)
- 13th International Conference on Document Analysis and Recognition (ICDAR'13)

BIBLIOGRAPHY

Abainia, K., Ouamour, S. & Sayoud, H. (2016). Effective language identification of forum texts based on statistical approaches. *Information Processing & Management*, 52(4), 491-512. doi: 10.1016/j.ipm.2015.12.003.

Ablavsky, V. & Stevens, M. R. (2003). Automatic feature selection with applications to script identification of degraded documents. *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*, pp. 750-754.

Aithal, P. K., Rajesh, G., Acharya, D. U. & Subbareddy, N. V. K. M. (2010). Text line script identification for a tri-lingual document. *Computing Communication and Networking Technologies (ICCCNT), 2010 International Conference on*, pp. 1-3.

Arabnejad, E., Moghaddam, R. F. & Cheriet, M. (2017). PSI: Patch-based script identification using non-negative matrix factorization. *Pattern Recognition*, 67, 328 - 339. doi: 10.1016/j.patcog.2017.02.020.

Azmi, M. S., Omar, K., Nasrudin, M. F., Ghazali, K. W. M. & Abdullah, A. (2011). *Arabic calligraphy identification for Digital Jawi Paleography using triangle blocks*. Conference Proceedings presented in Proceedings of the 2011 International Conference on Electrical Engineering and Informatics (pp. 1-5). doi: 10.1109/ICEEI.2011.6021785.

Bahmani, Z., Alamdar, F., Azmi, R. & Haratizadeh, S. (2010). Off-line Arabic/Farsi handwritten word recognition using RBF neural network and genetic algorithm. *Intelligent Computing and Intelligent Systems (ICIS), 2010 IEEE International Conference on*, 3, 352–357. doi: 10.1109/ICICISYS.2010.5658635.

Bataineh, B., Abdullah, S. N. H. S. & Omar, K. (2011). *Arabic calligraphy recognition based on binarization methods and degraded images*. Conference Proceedings presented in Proceedings of the 2011 International Conference on Pattern Analysis and Intelligent Robotics, ICPAIR 2011 (pp. 65–70). doi: 10.1109/ICPAIR.2011.5976913.

Bay, H., Ess, A., Tuytelaars, T. & Van Gool, L. (2008). Speeded-Up Robust Features (SURF). *Comput. Vis. Image Underst.*, 110(3), 346–359. doi: 10.1016/j.cviu.2007.09.014.

Ben Moussa, S., Zahour, A., Benabdelhafid, A. & Alimi, A. M. (2008). Fractal-based system for Arabic/Latin, printed/handwritten script identification. *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pp. 1-4.

Ben Moussa, S., Zahour, A., Benabdelhafid, A. & Alimi, A. M. (2010). New features using fractal multi-dimensions for generalized Arabic font recognition. *Pattern Recognition Letters*, 361-371. doi: 10.1016/j.patrec.2009.10.015.

Bilcu, E. B. & Astola, J. (2006). *A Hybrid Neural Network for Language Identification from Text*. Conference Proceedings presented in Machine Learning for Signal Processing, 2006. Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on (pp. 253-258).

Busch, A., Boles, W. W. & Sridharan, S. (2005). Texture for script identification. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(11), 1720-1732.

Cai, D., He, X., Han, J. & Huang, T. S. (2011). Graph Regularized Nonnegative Matrix Factorization for Data Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8), 1548-1560. doi: 10.1109/TPAMI.2010.231.

Chanda, S., Pal, S., Franke, K. & Pal, U. (2009). Two-stage Approach for Word-wise Script Identification. *Document Analysis and Recognition, 2009. ICDAR '09. 10th International Conference on*, pp. 926-930.

Chanda, S., Pal, U., Franke, K. & Kimura, F. (2010). Script Identification Survey; A Han and Roman Script Perspective. *Pattern Recognition (ICPR), 2010 20th International Conference on*, pp. 2708-2711.

Chanda, S. (2012). Font Identification – In Context of an Indic Script. *International Conference on Pattern Recognition*, (Icpr), 1655–1658.

Chawki, D. & Labiba, S. M. (2010). A texture based approach for Arabic Writer Identification and Verification. *2010 International Conference on Machine and Web Intelligence, ICMWI 2010 - Proceedings*, 115–120. doi: 10.1109/ICMWI.2010.5648130.

Choi, S. (2008, June). Algorithms for orthogonal nonnegative matrix factorization. *IEEE International Joint Conference on Neural Networks, 2008. IJCNN 2008*, pp. 1828-1832. doi: 10.1109/IJCNN.2008.4634046.

Choon-Ching, N. & Selamat, A. (2009). *Improved Letter Weighting Feature Selection on Arabic Script Language Identification*. Conference Proceedings presented in Intelligent Information and Database Systems, 2009. ACIIDS 2009. First Asian Conference on (pp. 150-154).

Chu, M., Diele, F., Plemmons, R. & Ragni, S. (2004). Optimality, computation, and interpretation of nonnegative matrix factorizations. *SIAM Journal on Matrix Analysis*, pp. –.

Cichocki, A. & ichi Amari, S. (2010). Families of Alpha- Beta- and Gamma- Divergences: Flexible and Robust Measures of Similarities. *Entropy*, 12(6), 1532-1568.

Cichocki, A., Zdunek, R. & Amari, S.-i. (2006). Csisar's Divergences for Non-negative Matrix Factorization: Family of New Algorithms. *Proceedings of the 6th International Conference on Independent Component Analysis and Blind Signal Separation*, (ICA'06), 32–39.

Cichocki, A., Lee, H., Kim, Y.-D. & Choi, S. (2008). Non-negative Matrix Factorization with α-divergence. *Pattern Recogn. Lett.*, 29(9), 1433–1440.

Cichocki, A., Cruces, S. & Amari, S.-i. (2011). Generalized Alpha-Beta Divergences and Their Application to Robust Nonnegative Matrix Factorization. *Entropy*, 13(1), 134–170.

da Silva, J. F. & Lopes, G. P. (2006). *Identification of Document Language is Not yet a Completely Solved Problem.* Conference Proceedings presented in Computational Intelligence for Modelling, Control and Automation, 2006 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on (pp. 212-212).

Daniels, P. & Bright, W. (1996). *The World's Writing Systems.* Oxford University Press.

Dhandra, B. V., Nagabhushan, P., Hangarge, M., Hegadi, R. & Malemath, V. S. (2006). Script Identification Based on Morphological Reconstruction in Document Images. *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, 2, 950-953. doi: 10.1109/ICPR.2006.1030.

Dhillon, I. S. & Sra, S. (2005). Generalized nonnegative matrix approximations with Bregman divergences. *In: Neural Information Proc. Systems*, pp. 283–290.

Ding, C., Li, T., Peng, W. & Park, H. (2006). Orthogonal Nonnegative Matrix T-factorizations for Clustering. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (KDD '06), 126–135. doi: 10.1145/1150402.1150420.

Ding, C. H. Q., Li, T. & Jordan, M. I. (2010). Convex and Semi-Nonnegative Matrix Factorizations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(1), 45–55. doi: 10.1109/TPAMI.2008.277.

Ding, X., Chen, L. & Wu, T. (2007). Character independent font recognition on a single Chinese character. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2), 195–204. doi: 10.1109/TPAMI.2007.26.

Donoho, D. & Stodden, V. (2003). When Does Non-Negative Matrix Factorization Give Correct Decomposition into Parts? *NIPS*, pp. 2004.

Elgammal, A. M. & Ismail, M. A. (2001). *Techniques for language identification for hybrid Arabic-English document images.* Conference Proceedings presented in Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on (pp. 1100-1104).

Fang, Y., Xue-Dong, T. & Bao-Lan, G. (2002). *An improved font recognition method based on texture analysis.* Conference Proceedings presented in Proceedings. International Conference on Machine Learning and Cybernetics (pp. 1726-1729 vol.4). doi: 10.1109/ICMLC.2002.1175331.

Farooq, F. & Govindaraju, V. (2007). *Language identification in historical Afghan manuscripts*. Conference Proceedings presented in Signal Processing and Its Applications, 2007. ISSPA 2007. 9th International Symposium on (pp. 1-4).

Farrahi Moghaddam, R., Farrahi Moghaddam, F. & Cheriet, M. (2012). A new framework based on signature patches, micro registration, and sparse representation for optical text recognition. *Information Science, Signal Processing and their Applications (ISSPA), 2012 11th International Conference on*, pp. 1259–1265.

Fragkou, P. (2014). Text Segmentation for Language Identification in Greek Forums. *Procedia - Social and Behavioral Sciences*, 147, 160-166. doi: 10.1016/j.sbspro.2014.07.140.

Garain, U. & Paquet, T. (2009). Off-line multi-script writer identification using AR coefficients. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 991–995. doi: 10.1109/ICDAR.2009.222.

Gomez, L., Nicolaou, A. & Karatzas, D. (2017). Improving patch-based scene text script identification with ensembles of conjoined networks. *Pattern Recognition*, 67, 85 - 96. doi: 10.1016/j.patcog.2017.01.032.

Gopakumar, R., Subbareddy, N. V., Makkithaya, K. & Acharya, U. D. (2010). Zone-based structural feature extraction for script identification from Indian documents. *Industrial and Information Systems (ICIIS), 2010 International Conference on*, pp. 420-425.

Ha, M.-H. H. M.-H., Tian, X.-D. T. X.-D. & Zhang, Z.-R. Z. Z.-R. (2005). Optical font recognition based on Gabor filter. *2005 International Conference on Machine Learning and Cybernetics*, 8(August), 18–21. doi: 10.1109/ICMLC.2005.1527799.

Hakkinen, J. & Jilei, T. (2001a). *n-gram and decision tree based language identification for written words*. Conference Proceedings presented in Automatic Speech Recognition and Understanding, 2001. ASRU '01. IEEE Workshop on (pp. 335-338).

Hakkinen, J. & Jilei, T. (2001b). *n-gram and decision tree based language identification for written words*. Conference Proceedings presented in Automatic Speech Recognition and Understanding, 2001. ASRU '01. IEEE Workshop on (pp. 335-338).

Hamza, A. B. & Brady, D. J. (2006). Reconstruction of reflectance spectra using robust nonnegative matrix factorization. *IEEE Transactions on Signal Processing*, 54(9), 3637-3642.

Hangarge, M. & Dhandra, B. V. (2008). Shape and Morphological Transformation Based Features for Language Identification in Indian Document Images. *Emerging Trends in Engineering and Technology, 2008. ICETET '08. First International Conference on*, pp. 1175-1180.

Hiremath, P. S., Shivashankar, S., Pujari, J. D. & Mouneswara, V. (2010a). Script identification in a handwritten document image using texture features. *Advance Computing Conference (IACC), 2010 IEEE 2nd International*, pp. 110-114.

Hiremath, P., Shivashankar, S., Pujari, J. & Kartik, R. (2010b). Writer identification in a handwritten document image using texture features. *Signal and Image Processing (ICSIP), 2010 International Conference on*, 110–114. doi: 10.1109/ICSIP.2010.5697457.

Howe, N. R. (2013). Document binarization with automatic parameter tuning. *International Journal on Document Analysis and Recognition (IJDAR)*, 16(3), 247–258. doi: 10.1007/s10032-012-0192-x.

Hoyer, P. O. (2002). Non-Negative Sparse Coding. *Proc. IEEE Workshop Neural Networks for Signal Processing*, pp. 557–565.

Hung-Ming, S. (2006). *Multi-Linguistic Optical Font Recognition Using Stroke Templates*. Conference Proceedings presented in 18th International Conference on Pattern Recognition (ICPR'06) (pp. 889-892). doi: 10.1109/ICPR.2006.824.

Jamjuntr, P. & Dejdumrong, N. (2009). Thai Font Type Recognition Using Linear Interpolation Analysis. *2009 Sixth International Conference on Computer Graphics, Imaging and Visualization*, 406–409. doi: 10.1109/CGIV.2009.72.

Khoddami, M. & Behrad, A. (2010). Farsi and Latin script identification using curvature scale space features. *Neural Network Applications in Electrical Engineering (NEUREL), 2010 10th Symposium on*, pp. 213-217.

Khosravi, H. & Kabir, E. (2010). Farsi font recognition based on Sobel-Roberts features. *Pattern Recognition Letters*, 31(1), 75–82. doi: 10.1016/j.patrec.2009.09.002.

Kim, Y.-D. & Choi, S. (2009, April). Weighted nonnegative matrix factorization. *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 1541-1544. doi: 10.1109/ICASSP.2009.4959890.

Kompass, R. (2007). A Generalized Divergence Measure for Nonnegative Matrix Factorization. *Neural Comput.*, 19(3), 780–791.

Kruengkrai, C., Srichaivattana, P., Sornlertlamvanich, V. & Isahara, H. (2005). *Language identification based on string kernels*. Conference Proceedings presented in Communications and Information Technology, 2005. ISCIT 2005. IEEE International Symposium on (pp. 926-929).

Kullback, S. & Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1), 79–86.

Laurberg, H. (2007). Uniqueness of Non-Negative Matrix Factorization. *Statistical Signal Processing, IEEE/SP Workshop on*, 0, 44-48. doi: 10.1109/SSP.2007.4301215.

Lee, C. W., Kang, H., Jung, K. & Kim, H. J. (2003). LNCS 2756 - Font Classification Using NMF. 470–477.

Lee, D. D. & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755), 788-791.

Lee, D. & Seung, S. (2001, Apr). Algorithms for Non-negative Matrix Factorization. *Advances in Neural Information Processing Systems 13*, pp. 556–562.

Li, L. & Tan, C. L. (2008). *Script identification of camera-based images*. Conference Proceedings presented in 2008 19th International Conference on Pattern Recognition (pp. 1-4). doi: 10.1109/ICPR.2008.4760965.

Li, L. & Zhang, Y.-J. (2007, Aug). Non-negative Matrix-Set Factorization. *Image and Graphics, 2007. ICIG 2007. Fourth International Conference on*, pp. 564-569. doi: 10.1109/I-CIG.2007.103.

Li Zhang,Yue Lu, C. L. T. (2004). Italic Font Recognition Using Stroke Pattern Analysis on Wavelet Decomposed Word Images [Conference Paper]. doi: 10.1109/icpr.2004.513.

Lidke, J., Thurau, C. & Bauckhage, C. (2010). The snippet statistics of font recognition. *Proceedings - International Conference on Pattern Recognition*, 1868–1871. doi: 10.1109/ICPR.2010.461.

Lin, C.-J. (2007). Projected Gradient Methods for Nonnegative Matrix Factorization. *Neural Comput.*, 19(10), 2756–2779. doi: 10.1162/neco.2007.19.10.2756.

Linlin, L. & Chew Lim, T. (2008). Script identification of camera-based images. *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pp. 1-4.

Ljubešić, N., Mikelić, N. & Boras, D. (2007). *Language identification: How to distinguish similar languages?* Conference Proceedings presented in Information Technology Interfaces, 2007. ITI 2007. 29th International Conference on (pp. 541-546).

Lowe, D. G. (1999). Object Recognition from Local Scale-Invariant Features. *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, (ICCV '99), 1150–. Consulted at http://dl.acm.org/citation.cfm?id=850924.851523.

Lu, S. & Chew Lim, T. (2008a). Script and Language Identification in Noisy and Degraded Document Images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(1), 14-24.

Lu, S. & Chew Lim, T. (2008b). Script and Language Identification in Noisy and Degraded Document Images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(1), 14-24.

Luqman, H., Mahmoud, S. A. & Awaida, S. (2014). KAFD Arabic font database. *Pattern Recognition*, 47(6), 2231–2240. doi: 10.1016/j.patcog.2013.12.012.

Manna, S. L., a.M. Colia & Sperduti, a. (1999). Optical font recognition for multi-font OCR and document processing. *Proceedings. Tenth International Workshop on Database and Expert Systems Applications. DEXA 99*, 549-553. doi: 10.1109/DEXA.1999.795244.

Marinai, S., Miotti, B. & Soda, G. (2010). Bag of Characters and SOM Clustering for Script Recognition and Writer Identification. *Pattern Recognition (ICPR), 2010 20th International Conference on*, pp. 2182-2185.

Miao, X.-F. M. X.-F., Tian, X.-D. T. X.-D. & Guo, B.-L. G. B.-L. (2002). Individual character font recognition based on guidance font. *Proceedings. International Conference on Machine Learning and Cybernetics*, 4(November), 4–5. doi: 10.1109/ICMLC.2002.1175328.

Moussa, S. B., Zahour, A., Benabdelhafid, A. & Alimi, A. M. (2008, Dec). Fractal-based system for Arabic/Latin, printed/handwritten script identification. *2008 19th International Conference on Pattern Recognition*, pp. 1-4. doi: 10.1109/ICPR.2008.4761838.

Nafchi, H. Z., Moghaddam, R. F. & Cheriet, M. (2014). Phase-Based Binarization of Ancient Document Images: Model and Applications. *IEEE Transactions on Image Processing*, 23(7), 2916-2930. doi: 10.1109/TIP.2014.2322451.

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *Systems, Man and Cybernetics, IEEE Transactions on*, 9(1), 62-66.

Paatero, P. & Tapper, U. (1994). Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2), 111–126. doi: 10.1002/env.3170050203.

Padma, M. C. & Vijaya, P. A. (2009). Monothetic Separation of Telugu, Hindi and English Text Lines From a Multilingual. *SMC*, pp. 4870-4875.

Padma, M. C., Vijaya, P. A. & Nagabhushan, P. (2009). *Language Identification from an Indian Multilingual Document Using Profile Features*. Conference Proceedings presented in Computer and Automation Engineering, 2009. ICCAE '09. International Conference on (pp. 332-335).

Pal, U. & Chaudhuri, B. B. (2001). Automatic identification of English, Chinese, Arabic, Devnagari and Bangla script line. *Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on*, pp. 790-794.

Pal, U., Sinha, S. & Chaudhuri, B. B. (2003). Multi-script line identification from Indian documents. *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*, pp. 880-884.

Pan, J. & Tang, Y. (2011). A rotation-robust script identification based on BEMD and LBP. *Wavelet Analysis and Pattern Recognition (ICWAPR), 2011 International Conference on*, pp. 165–170.

Pan, W. M., Suen, C. Y. & Bui, T. D. (2005). Script identification using steerable Gabor filters. *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on*, pp. 883-887 Vol. 2.

Peake, G. S. & Tan, T. N. (1997). *Script and language identification from document images*. Conference Proceedings presented in Document Image Analysis, 1997. (DIA '97) Proceedings., Workshop on (pp. 10-17).

Pourasad, Y., Hassibi, H. & Banaeyan, M. (2011). *Farsi font recognition based on spatial matching*. Conference Proceedings presented in 2011 18th International Conference on Systems, Signals and Image Processing (pp. 1-4).

Ramanathan, R., Ponmathavan, S., Thaneshwaran, L., , A. S. N., Valliappan, N. & Soman, K. P. (2009a). Tamil Font Recognition Using Gabor Filters and Support Vector Machines [Conference Paper]. doi: 10.1109/act.2009.156.

Ramanathan, R., Soman, K., Thaneshwaran, L., Viknesh, V., Arunkumar, T. & Yuvaraj, P. (2009b). A Novel Technique for English Font Recognition Using Support Vector Machines. *2009 International Conference on Advances in Recent Technologies in Communication and Computing*, 1(1), 766-769. doi: 10.1109/ARTCom.2009.89.

Rezaee, H., Geravanchizadeh, M. & Razzazi, F. (2009). Automatic language identification of bilingual English and Farsi scripts. *Application of Information and Communication Technologies, 2009. AICT 2009. International Conference on*, pp. 1-4.

Roy, K., Alaei, A. & Pal, U. (2010). Word-Wise Handwritten Persian and Roman Script Identification. *Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on*, pp. 628-633.

Sandler, R. & Lindenbaum, M. (2011). Nonnegative Matrix Factorization with Earth Mover's Distance Metric for Image Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8), 1590-1602.

Schreyer, A., Suda, P. & Maderlechner, G. (1999). A formal approach to textons and its application to font style detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1655, 72–83. doi: 10.1007/3-540-48172-9_7.

Selamat, A. & Ng Choon, C. (2008). *Arabic Script Documents Language Identifications Using Fuzzy ART*. Conference Proceedings presented in Modeling & Simulation, 2008. AICMS 08. Second Asia International Conference on (pp. 528-533).

Selamat, A., Ng Choon, C. & Mikami, Y. (2007). *Arabic Script Web Documents Language Identification Using Decision Tree-ARTMAP Model*. Conference Proceedings presented in Convergence Information Technology, 2007. International Conference on (pp. 721-726).

Shashua, A. & Hazan, T. (2005). Non-negative tensor factorization with applications to statistics and computer vision. *In Proceedings of the International Conference on Machine Learning (ICML)*, pp. 792–799.

Shi, H. S. H. & Pavlidis, T. (1997). Font recognition and contextual processing for more accurate text recognition. *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, 1, 39–44. doi: 10.1109/ICDAR.1997.619810.

Shijian, L. & Chew Lim, T. (2007). Automatic Detection of Document Script and Orientation. *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, 1, 237-241.

Singhal, V., Navin, N. & Ghosh, D. (2003). Script-based classification of hand-written text documents in a multilingual environment. *Research Issues in Data Engineering: Multilingual Information Management, 2003. RIDE-MLIM 2003. Proceedings. 13th International Workshop on*, pp. 47-54.

Slimane, F., Kanoun, S., Alimi, A. M., Ingold, R. & Hennebert, J. (2010). Gaussian mixture models for arabic font recognition. *Proceedings - International Conference on Pattern Recognition*, 2174–2177. doi: 10.1109/ICPR.2010.532.

Slimane, F., Kanoun, S., Hennebert, J., Alimi, A. M. & Ingold, R. (2013). A study on font-family and font-size recognition applied to Arabic word images at ultra-low resolution. *Pattern Recognition Letters*, 34(2), 209–218. doi: 10.1016/j.patrec.2012.09.012.

Smaragdis, P. (2007). Convolutive Speech Bases and Their Application to Supervised Speech Separation. *IEEE Transactions on Audio, Speech & Language Processing*, 15(1), 1-12.

Spitz, A. L. (1997a). Determination of the script and language content of document images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(3), 235-245.

Spitz, A. L. (1997b). Determination of the script and language content of document images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(3), 235-245.

Sun, H. M. (2006). Multi-linguistic optical font recognition using stroke templates. *Proceedings - International Conference on Pattern Recognition*, 2, 889–892.

Takçı, H. & Güngör, T. (2012). A high performance centroid-based classification approach for language identification. *Pattern Recognition Letters*, 33(16), 2077-2084. doi: 10.1016/j.patrec.2012.06.012.

Tian, J. & Suontausta, J. (2003). *Scalable neural network based language identification from written text*. Conference Proceedings presented in Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on (pp. I-48-51 vol.1).

Wang, Y.-X. & Zhang, Y.-J. (2013). Nonnegative Matrix Factorization: A Comprehensive Review. *Knowledge and Data Engineering, IEEE Transactions on*, 25(6), 1336-1353. doi: 10.1109/TKDE.2012.51.

Wang, Y. & Jia, Y. (2004). Fisher non-negative matrix factorization for learning local features. *In Proc. Asian Conf. on Comp. Vision*, pp. 27–30.

Wood, S. L., Xiaozhong, Y., Krishnamurthi, K. & Dang, L. (1995). *Language identification for printed text independent of segmentation*. Conference Proceedings presented in Image Processing, 1995. Proceedings., International Conference on (pp. 428-431 vol.3).

Xi, Y. & Wenxin, L. (2010). *An N-Gram-and-Wikipedia joint approach to Natural Language Identification*. Conference Proceedings presented in Universal Communication Symposium (IUCS), 2010 4th International (pp. 332-339).

Yang, Z. & Oja, E. (2010). Linear and Nonlinear Projective Nonnegative Matrix Factorization. *Neural Networks, IEEE Transactions on*, 21(5), 734-749. doi: 10.1109/TNN.2010.2041361.

Yong, Z., Tieniu, T. & Yunhong, W. (2001). Font recognition based on global texture analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10), 1192-1200. doi: 10.1109/34.954608.

Yoo, J. & Choi, S. (2010). Orthogonal nonnegative matrix tri-factorization for co-clustering: Multiplicative updates on Stiefel manifolds. *Inf. Process. Manage.*, 46(5), 559-570.

Yuan, Z. & Oja, E. (2005). Projective Nonnegative Matrix Factorization for Image Compression and Feature Extraction. *SCIA*, 3540(Lecture Notes in Computer Science), 333-342.

Yuemei, R., Yangning, Z., Ying, L., Jianyu, H. & Jianjiang, H. (2011). A Space Target Recognition Method Based on Compressive Sensing. *Image and Graphics (ICIG), 2011 Sixth International Conference on*, pp. 582-586.

Zhang, L., Lu Y. & Tan, C. L. (2004). Italic font recognition using stroke pattern analysis on wavelet decomposed word images. *Proceedings - International Conference on Pattern Recognition*, 4, 835–838. doi: 10.1109/ICPR.2004.1333902.

Zhou, L., Ping, X. J., Zheng, E. G. & Guo, L. (2010). Script identification based on wavelet energy histogram moment features. *Signal Processing (ICSP), 2010 IEEE 10th International Conference on*, pp. 980-983.

Zhu, G., Yu, X., Li, Y. & Doermann, D. (2009). Language identification for handwritten document images using a shape codebook. *Pattern Recognition*, 42(12), 3184 - 3191. doi: 10.1016/j.patcog.2008.12.022. New Frontiers in Handwriting Recognition.

Zhu, Y., Tan, T. & Wang, Y. (2001). Font recognition based on global texture analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10), 1192–1200. doi: 10.1109/34.954608.

Zramdini, A. & Ingold, R. (1998). Optical font recognition using typographical features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8), 877–882. doi: 10.1109/34.709616.