T.R.

GEBZE TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIEDSCIENCES

A NEW SEGMENTATION APPROACH TO UIGHUR OPTIC CHARACTER RECOGNITION

MEMTIMIN MAHMUT

A THESIS SUBMITTED FOR THE DEGREE OF

MASTER OF SCIENCE

DEPARTMENT OF COMPUTER ENGINEERING

GEBZE

2020

T.R.

GEBZE TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

A NEW SEGMENTATION APPROACH TO UIGHUR OPTIC CHARACTER RECOGNITION

MEMTİMİN MAHMUT

A THESIS SUBMITTED FOR THE DEGREE OF

MASTER OF SCIENCE

DEPARTMENT OF COMPUTER ENGINEERING

THESIS SUPERVISOR

ASSİST. PROF. DR. YAKUP GENÇ

GEBZE

2020

T.C.

GEBZE TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

UYGURCA KARAKTER TANİMADA YENİ BİR SEGMENTASYON YÖNTEMİ

MEMTİMİN MAHMUT

YÜKSEK LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI

DR. ÖĞR. ÜYESİ YAKUP GENÇ

GEBZE

2020

GEBZE
TEKNİK ÜNİVERSİTESİ

YÜKSEK LİSANS JÜRİ ONAY FORMU

GTÜ Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 29/01/2020 tarih ve 2020/07 sayılı kararıyla oluşturulan jüri tarafından 17/02/2020 tarihinde tez savunma sınavı yapılan Memtimin...MAHMUT'un tez çalışması Bilgisayar Mühendisliği Bölümü Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE
(TEZ DANIŞMANI) : Dr.Öğr.Üyesi Yakup GENÇ

ÜYE : Prof. Dr.Fatih Erdoğan SEVİLGEN

ÜYE : Dr.Öğr.Üyesi Ayşe Betül OKTAY

ONAY

Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun
........./....../........ tarih ve ..../....... sayılı kararı.

SUMMARY

Optic character recognition (OCR) is software that translates the image of writing to editable and researchable text in ASCII or Unicode format. OCR systems have possessed intensive research value and commercial exploitation because of its attribute to convert the text data from conventional media into electronic media. It's used place is also widened from several kinds of document processing in the office to supplement the attached program in machine vision research and developments. The first concept of the idea of OCR was put forward in 1929 when the modern computer had not been created yet, was mechanical equipment which involved a photo detector, obtained the first patent on OCR. With the invention of digital computers in the middle of 1940, the OCR system absorbing more interesting in scientific (pattern recognition) research and commercial products area. A commercial OCR was available in 1950. In 2001, OCR systems began to provide online service on the internet and it is already free used techniques or software nowadays. However, those are only limited to OCR for a non-cursive script such as Latin script. For example, OCR systems for Latin, Japanese and Chinese are well developed because of the character of them are isolated which makes their OCR techniques easier to develop. The OCR techniques on the other language scripts including Arabic and Uighur have not been well developed compared to the OCR on Latin script. Arabic OCR, after publishing the first paper in 1975, a large number of research papers and technical reports have appeared and many new techniques have been developed, but it is still an open research field in OCR techniques due to the technical difficulties included by the cursive nature of Arabic script. Uighur alphabets, which were produced based on Arabic script, are the same difficulties and handicappers in research and developing Uighur OCR. In this work, we put up with a segmentation method in the Uighur OCR as well as Arabic OCR that it is easy at calculation design and time complexity. We have observed the sufficiency and productivity of this method by experiment. Then apply the deep learning approach in the classification stage to recognize the three consecutive characters as a unit. Meanwhile, we might assume that this segmentation method is also available to another Arabic script-based language.

Keywords: Optical Character Recognition (OCR), Uighur OCR, Arabic OCR, Character Segmentation, Machine Learning, Deep Learning.

ÖZET

Optik Karakter Tanıma (OKT) belgere simdeki harfleri tanıma ve dijital metne çevirmektir. Karakter tanıma sistemleri konvensiyonel medyadan elektronik medyaya çevirmede yoğun kullanım alanı bulmuştur. İlk karakter tanıyıcı 1929 yılında mekanik makine olarak tasarlanmış ve patenti alınmıştır. 1940'lı yıllarında modern bilgisayarın icat edilmesiyle birlikte, otomatik karakter tanıma makinesi bilimsel araştırma ve ticari üretim alanında yoğun ilgi çekmiş ve 1950'li yıllarında ticari amaçla üretilen tanıma makineleri piyasaya girmiştir. 2001 yılında karakter tanıma servisi internet üzerinden temin edilmeye başlamışken, şimdi artık ücretsiz kullanabilmektedir. Ancak bu sistemler Latin alfabesi gibi bitişik olamayan karakterler üzerindeki tanıma problemleri için iyi çalışmaktadır. Bitişik yazılar üzerindeki tanıma problem daha az çalışılmış olup bazı dil karakterleri için henüz olgunlaşmamıştır. Mesela Latince, Japonca ve Çince karakter tanıma sistemleri karakterlerin bitişik olmayan izole özelliğinden dolayı olgunlaşmış sistemler olarak sayılmaktadır. Arapça karakter tanıma ise ilk olarak 1975 yılında çalışılmış olmakla birlikte hala aktif bir araştırma konusudur. Arapçanın böyle olmasının temel nedeni ise bitişik olarak yazılan veya basılan Arapça yazının segmentasyon işleminin başarılı olarak yapılamamasından kaynaklanmaktadır. Uygurca da Arap alfabesi temelindeki harflerle yazılan bir dildir ve bu konuda ilk çalışma 1996 yılında yapılmış olmakla birlikte mevcut system performansları Latin alfabesi düzeyinde değildir. Bunun nedeni de segmentasyon işleminin çok başarlı olamamasındandır. Bu çalışmada, Uygurca yazının segmentasyon işleminde gözlemlediğimiz bazı özellikleri ortaya koymuş ve segmentasyon işlemini daha kolay gerçekleştirime ve hatayı azaltma bakımından katkı sağladığını yaptığımız deneylerle gözlemlemiş bulunuyoruz. Sonraki sınıflandırma aşamasında, derin öğrenme yöntemini üç serikarakterli resime uygulama yoluyla tanıma gerçekleştirmiş ve deneylerle bunun performansı artırdığı görülmüştür. Aynı zamanda bu çalışmamız Arap alfabesi temelinde yazılan başka dil yazıları için de geçerli olacaktır.

Anahtar Kelimeler: **Optik Karakter Tanıma (OKT), Uygurca OKT, Arapça OKT,Karakter Segmentasyon Yöntemi, Makine Öğrenmesi, Derin Öğrenme**.

ACKNOWLEDGEMENTS

I thank my advisor, Assist. Prof. Dr Yakup Genc, for his guidance, inspiration, mental supports and patience throughout the process of researching and writing this work.

TABLE of CONTENTS

LISTof ABBREVIATIONS and ACRONYMS

Abbreviations and Acronyms	**Explanations**
$P(T)$	Cumulative probability function
$p(g)$	Probability mass function
$m_f(T), m_b(T)$	Mean of foreground and background
$\delta_f(T), \delta_b(T)$	Foreground and background Standard deviation
δ	Variance
$H_f(T), H_f(T)$	Entropy of the foreground and background
$\mathrm{P}\ (\mathrm{g}, \bar{\mathrm{g}})$	The histogram of the gray image…
$\bar{\mathrm{g}}$	The average grey value…
$(\mathrm{T}, \overline{\mathrm{T}})$	Threshold pair
$\mathrm{P}\ (\mathrm{T}, \overline{\mathrm{T}})$	The cumulative distribution of the threshold pair
$b_1(T), b_2(T)$	Approximated function used for foreground and background
θ	Skewness of the text image
OCR	Optic character recognition
UNICODE	The computing industry standard for the non-English character set(1991).
ASCII	American standard code for information interchange. It is an encoding standard for the English character set (1963).
ASMO	Arab standardization and metrology organization (1982).
AI	Artificial intelligence
PR	Pattern recognition
ANN	Artificial neural network
CNN	Convolutional neural network
RNN	Recurrent neural network
MLP	Multilayer perception
RTF	Random forest
SVM	Support vector machine
KNN	K-Nearest Neighbors
IPA	International Phonetic Alphabet
MST	Minimal Spanning Tree

BFS	Black field separators
WFS	White field separators
pBLS	Potential baseline segments
BLS	Baseline segments
LAG	Line adjacency graph
MCR	Minimum Covering Run
SHT	Standard Hough Transform
GHT	Generalized Hough Transform
WLAR	Word level Arabic recognition
HMM	Hidden Markov Models
LeNet	It is a classic CNN Architecture
NLP	Natural language processing
PCA	Principle component analysis
GTU	Gebze technical university
OKT	Optik Karakter Tanıma (Turkish translation of Optic character recognition)

LIST of FIGURES

LIST of TABLES

1. INTRODUCTION

Optical character recognition (OCR) extracts from an image a text in ASCII or Unicode format with correct punctuations. OCR is widely used successfully in many applications for non-cursive alphabets. For some alphabets and languages including Uighur language, OCR techniques are not as strong as for non-cursive script alphabets such as Latin, Chinese, Japanese alphabets [1].

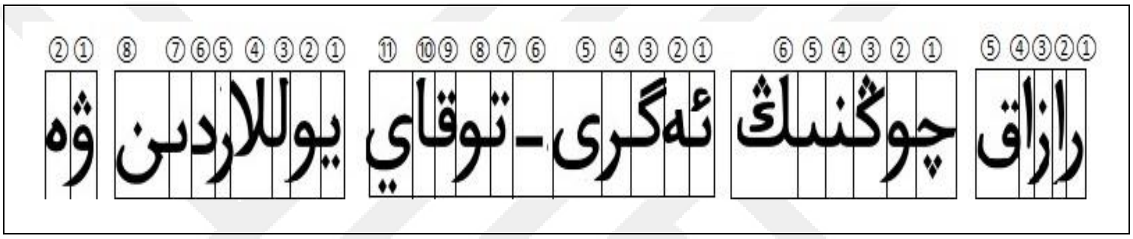
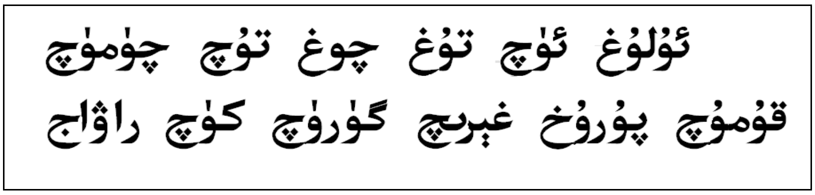


Figure 1.1: A sample line of text in the Uighur alphabet. Each character may be connected from one or both sides; it is a reason for cursiveness. However, some of them can be connected from one side only; the right, which causes a discontinuity of cursiveness. But some of those discontinuities are ambiguous from the vertical point of view, which appeared the overlap. The shape of the letter is context-sensitive, depending on its location[2]. For example, the last letter in the third word and the first letter in the fourth word are the same character but their shapes are different.

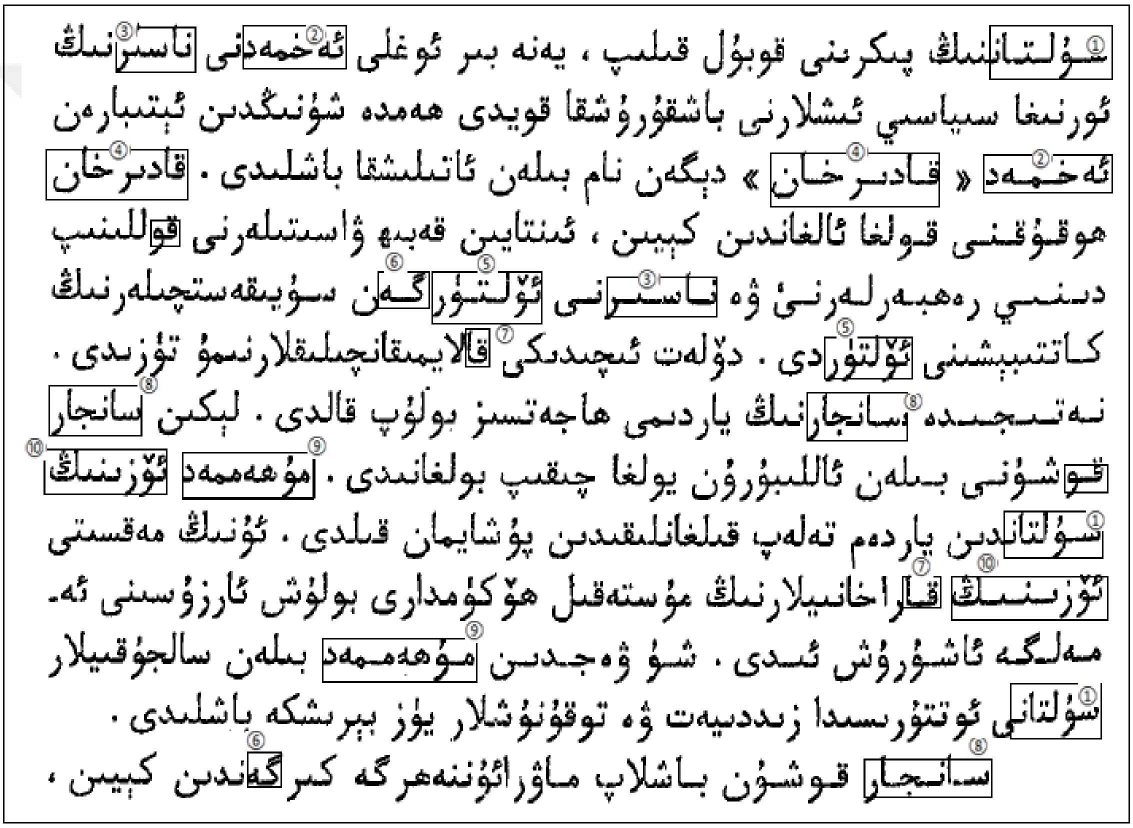
It is a difficult problem and open research field because of the cursive nature script and overlapping appearance of the Uighur written text, even in the printed text. The difficulty stems from the fact that the Uighur writing uses cursive scripts in which the varying length letters can be connected from one or both sides. Furthermore, words and sub-words are separated (or segmented) by subjective rules. These rules largely depend on the letter positions in the text. For example, consider the text given in Figure 1.1, all letters in the first word and the fifth word are not connected from any side, the 4th and 5th letters in the second word are connected from both right and left side, and the other letters are connected from the right or left side. In this example, overlap occurs between the 1st and 2nd letters in the first word, etc. In Figure 1.2 we show the ligature in the Uighur writing. The Figure 1.3 is the examples of the unbalanced length of the Uighur letter that each letter has no stable width even the same word in different places.



ئۇلۇغ ئۈچ تۇغ چوغ تۇچ چۈمۈچ

قۇمۇچ پۇرۇخ غىرىچ گۈرۈچ كۈچ راۋاج

Figure 1.2: Ligature in the Uighur writing. These letters overlap vertically and touching each other



سۇلتاننىڭ پىكرىنى قوبۇل قىلىپ ، يەنە بىر ئوغلى ئەخمەدنى ناسىرنىڭ
ئورنىغا سىياسىي ئىشلارنى باشقۇرۇشقا قويدى ھەمدە شۇنىڭدىن ئېتىبارەن
ئەخمەد « قادىرخان » دېگەن نام بىلەن ئاتىلىشقا باشلىدى . قادىرخان
ھوقۇقىنى قولغا ئالغاندىن كېيىن ، ئىنتايىن قەبىھ ۋاستىلەرنى قوللىنىپ
دىنىي رەھبەرلەرنى ۋە ناسىرنى ئۆلتۈرگەن سۇيىقەستچىلەرنىڭ
كاتتىبېشىنى ئۆلتۈردى . دۆلەت ئىچىدىكى قالايمىقانچىلىقلارنىمۇ تۈزىدى .
نەتىجىدە سانجارنىڭ ياردىمى ھاجەتسىز بولۇپ قالدى . لېكىن سانجار
قوشۇنى بىلەن ئاللىبۇرۇن يولغا چىقىپ بولغانىدى . مۇھەممەد ئۆزىنىڭ
سۇلتاندىن ياردەم تەلەپ قىلغانلىقىدىن پۇشايمان قىلدى . ئۇنىڭ مەقسىتى
ئۆزىنىڭ قاراخانىيلارنىڭ مۇستەقىل ھۆكۈمدارى بولۇش ئارزۇسىنى ئە-
مەلگە ئاشۇرۇش ئىدى . شۇ ۋەجىدىن مۇھەممەد بىلەن سالجۇقىيلار
سۇلتانى ئوتتۇرىسىدا زىددىيەت ۋە توقۇنۇشلار يۈز بېرىشكە باشلىدى .
سانجار قوشۇن باشلاپ ماۋرائۇننەھرگە كىرگەندىن كېيىن ،

Figure 1.3: The Unbalanced length of the Uighur letters. The lengths of the first three same words are 77, 62, 58 pixels, the lengths of the second two same words are 65, 82 pixels, the lengths of the third are 51, 70 pixels, the lengths of the fourth are 91, 107 pixels, the lengths of fifth are 69, 53 pixels, the length of the sixth are 31, 21 pixels, the lengths of the seventh are 14, 25 pixels, the lengths of the eighth are 65, 60, 80 pixels, the lengths of the ninth are 82, 98 pixels, the lengths of the tenth are 80, 99 pixels.

OCR has some research value and commercial exploitation in related fields such as artificial intelligence, pattern recognition, and computer vision. It's used place is also widened from several kind of document processing in the office to supplement the attached program in machine vision researches and developments. In

other words, the OCR also has somewhat used value in research and development in related fields.

The functional core of Uighur or Arabic OCR system may be divided into two operation stages: the character segmentation and classification. From the perspective of the classification techniques, to classify or recognize the segmented pictures of character is not already complex work for our present technology in computer vision and machine learning technology, however, because of the segmentation point at the Uighur and Arabic word or sub-word is a subjective point that very hard to calculate precisely, the character segmentation is still the challenge and obstacle for Uighur and Arabic OCR research, and the results of segmentation step would determine or greatly affect the accuracy of classification.

Most of the advanced approaches to the segmentation so far are not providing more calculation details [3]-[4], low correction of the results [5]-[6] and so on. Similarly, most of the classification techniques used in Uighur OCR is conventional machine learning techniques [7]-[8] that depend on the feature extraction method.

In this work, we focused on the segmentation process and presented a new approach to calculate the word and character segmentation point more easily, correctly and clearly. In the word segmentation, we get the width of the space between the words and space between the subwords (or space inside the word) for several hundred text line images, and then calculate the maximum value of the space histogram as a threshold value to the word and sub-word space difference. The encountered space whose width is larger than a threshold is word separated points otherwise is sub-word separated points.

In the character segmentation, we search the definite, continued and constant value from the vertical projection profile of the sub-word image and get the center of the x coordinate to the constant length as the character segmentation points. The definite and constant value is calculable that we could get it from several line text images and use it for all page or whole document while the font size is stable.

At the classification stage, we proposed a consecutive three-block character as

a recognition unit to remedy the potential remaining problem in the previous character segmentation stage and applied the deep learning approach to classify or identify the character image without extracting features by hand. Thus, we manage to improve the overall performance and efficiency of the system. In our approach, after determining the character segmentation points, we cut the three consecutive characters as a whole unit and represent them as middle characters instead of cutting them one by one as a single character only. Finally, the cropped raw images are fed into the classifier without extracting any features in advance. We use LeNet-5 [9] as the classifier, which is the architecture of convolutional neural networks (CNN) that extracts features from raw input image by training with a backpropagation algorithm.

The experimental results show that the proposed word and character segmentation approaches get 95.68% and 94.74% correctness respectively. Comparing the single character recognized unit with a 91.98% recognition rate, the proposed three-block character unit more adept to classify the characters series in the word with a 99.33% recognition rate for isolated characters. Compared with traditional learning algorithms such as random decision forests [10] and XgBoost[11], the deep learning algorithm is more efficient and convenient to use especially in scanned noisy image of a text. We evaluate our experiment result both on scanned paper with a stained or noisy and synthesized image with salt pepper noise added datasets.

The remainder of the thesis is organized as the following. In Chapter 2, we provide a background to the Uighur optical character recognition system by describing the problem in every operation stage of Uighur OCR as well as the morphological characteristics of Uighur writing. Then the comprehensive up-to-date review of researches and technical reports related to all processing steps in the Uighur OCR system. In this section, we present from two to three representative or necessary descriptions of algorithms or main contents of approach for each category. Finally, we present the new methods in those solutions, description of the datasets to evaluate those methods and performance evaluation for OCR applications and approaches in those operation stages. In Chapter 3, we describe the design and implementation of our proposed solution related to the Uighur OCR system. In Chapter 4, we describe the experiment and results by presenting the dataset used to evaluate the proposed solution and the cross-validation experiment detail. Then, we gave the results, analyzed the reason, discuss the handled and ignored aspects. In Chapter 5, we finish by a brief conclusion including the methods, summary and feature work.

2. OCR FOR UIGHUR LANGUAGE

Optical character recognition (OCR) systems extract text from images obtained through scanning or photographing of documents and get a computer representation output of this text. More than one definition and expression have been concerned with the OCR concept. Some of them are: is a pattern recognition application with the ultimate aim of simulating the human reading capabilities of both machine-printed and handwritten cursive text [1], transform human-readable characters to machine-readable codes (ASCII for Latin and SMO for Arabic as well as Unicode for others) [12], is the process of converting a raster image representation of a document, e.g. a machine printed or handwritten text scanned by a document scanner, into a computer processable format, such as ASCII code [13], is machine simulation of human reading [14]-[15], is machine replication of human reading [16], is the converting the data from conventional media into the new electronic media [2] and so on.

OCR is a field of research in artificial intelligence, pattern recognition, and computer vision. The artificial intelligence (AI) is the science and engineering of making intelligent machines [17] in thinking, acting, and learning aspects. The pattern recognition (PR) is the assignment of a physical object or event to one of several prespecified categories [18]. It is concerned with the classification of objects into categories, especially by machine [19]. In this definition, the pattern is an object, process, and event that can be given a name. The computer vision is the science and technology of machines that see [20], it is an interdisciplinary field and it generates mathematical models or computer symbols from images.

OCR has somewhat scientific research (pattern recognition) value and commercial exploitation in related fields. The origins of the character recognition system can be found in 1870 with the invention of the retina scanner by Charles R. Carey and the invention of sequential scanner by Nipkow in 1890 which was a breakthrough for modern television and reading machines. The character recognition first appeared as an aid to the visually handicapped and the first successful attempts were made by the Russian scientist Tyurin in 1900 [21]. In 1929 Tausheck in Germany obtained a patent on OCR [22] in which use the template matching. In

1933 Handel in the U.S did the same [23]. The two patents are the first OCR concepts as published in the literature. The modern version of OCR appeared in the middle of the 1940s with the development of digital computers [24]. Since then many scientists and engineers have started the researchers on OCR not only because of the very challenging nature of the problem but also because it improves human-machine interaction in many applications [13]. The commercial OCR systems were available in 1950 and the researchers for Chinese OCR were started in 1960 [2]. The first research paper has reported in 1966 for the Chinese language [25] and in 1975 for the Arabic language [26].

In terms of acquiring their input, character recognition system can be divided into two types: online recognition system and off-line recognition system [27], each having its own hardware and algorithms.

In the online recognition system, the characters are recognized as it is being written while offline recognition is performed after writing, typing or printing is completed. The online recognition system has to employ special input equipment such as an electronic tablet with a stylus pen. The electronic tablet captures the co-ordinate data of the pen-tip movement and an indication of pen-up and pen-down [28].

Offline recognition systems may acquire the image of text using a video camera or scanner [29]–[32]. Further details about the difference between the online and the offline character recognition system are given by [33].

Since the first research paper for Arabic OCR in 1975 [26], for Uighur OCR in 1996 [4], a large number of research papers and technical reports have appeared and many new techniques have been developed. According to their approaches in tackling word segmentation, those techniques can be categorized into two strategies: the holistic strategies and analytical strategies. In holistic strategies the recognition is globally performed on the whole representation of words, whereas in analytical strategies, the words are considered as sequences of small size units and the recognition is performed at those units, which can be graphemes, segments, pseudo-letter, etc. [29],[34]-[36].

There are three writing styles according to the complexity of writing: typewriting or machine-printed, typeset, and handwritten. The typewriting or machine-printed style is a computer-generated style and it is the simplest among all styles because of the uniformity in writing a word. The typeset is usually more difficult than a machine-printed style because of the existence of overlap and ligature. Overlaps occur when two or more letters overlap vertically without touching (shown in Figure 1.1) and ligatures occur when two or more letters overlap vertically and touch (shown in Figure 1.2). The handwritten is the most difficult style because of the variation in character shape even if it is rewritten by the same person. Handwritten style may be further classified into a scribe, personal and decorative. The scribe is more carefully written than the personal handwriting style that represents the daily usage of the Arabic alphabet as well as the Uighur alphabet by individuals. Few people be able to perform an exquisite scribe handwritten script. Decorative handwriting is normally used for adornment purposes [2].

Relative to their approach in classifying the segmented characters, the classification of machine learning methods fall into either the conventional neural network such as MLP(multilayer perception), SVM(support vector machine), RTF(random forest), KNN(K-Nearest Neighbors),and representation neural networks such as deep learning including CNN (convolutional neural network), RNN (Recurrent Neural Network).

In this work, we concentrate on the offline, typeset Uighur text, analytical approach strategy, and deep learning classification methods.

2.1. The Morphological Characteristics of Uighur Written Text

The Uighur script has been created based on Arabic script and the Arabic script different from Latin script in essence, especially in justification and cursiveness. Opposed to Latin script, Arabic script is written from right to left and starts from the right-most position of the page towards the left in a cursive way, even in the machine printed text [2]. We describe the morphological characteristics of the Uighur writing by presenting the differences and similarities between the Latin script and Arabic script, as well as between the Uighur alphabet and the Arabic alphabet.

The main differences between the Latin script and Arabic scripture are:

i) The representation of vowels. Since Arabic utilizes various diacritical markings and Latin script employs 5 sonants instead of diacritics.

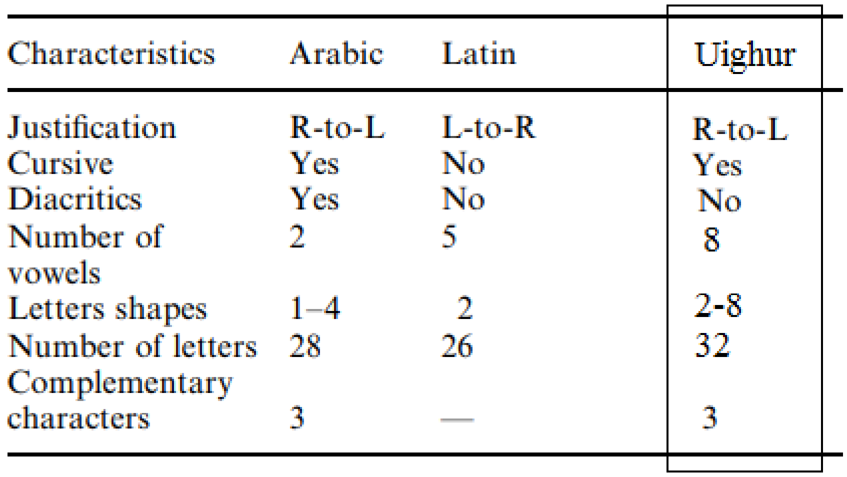
ii) Letter shape. A Latin letter has two possible shapes, capital and small, whereas the Arabic letter might have up to four different shapes depending on its relative location in the text.

iii) Complementary character. Different Arabic characters may have the same shape and are distinguished from only by the addition of complementary characters which are the portion of characters that are needed to complement an Arabic character. The complementary characters are a dot, a group of dots or a zigzag. Those may appear on, above, or below the baseline and positioned differently. But the Latin script does not use complementary characters [14].

These differences and similarities are outlined in Table 2.1 where the columns of Arabic and Latin are cited from [14].

As to be created from Arabic script, the Uighur alphabet, which is a little modified version of the Turkish alphabet in Arabic script, has involved most of the characteristics of the Arabic alphabet; meanwhile, the Uighur alphabet also has some features that distinguished from Arabic.

Table 2.1: The Comparison of Uighur, Arabic and Latin Scripts. The justification, cursive and complementary characters are the same as the Arabic script. The Uighur script has no diacritics, which is different from the Arabic script.



Characteristics	Arabic	Latin	Uighur
Justification	R-to-L	L-to-R	R-to-L
Cursive	Yes	No	Yes
Diacritics	Yes	No	No
Number of vowels	2	5	8
Letters shapes	1–4	2	2-8
Number of letters	28	26	32
Complementary characters	3	—	3

We describe the morphological characteristics of the Uighur writing in terms of similarities and differences between the Arabic and Uighur writing considering the appearance and writing styles.

The main similarity of them could be presented in several aspects.

i) Cursive script natures. Because the characters are connected to each other from one or both sides.

ii) The characters in word are connected on the baseline.

iii) Space is used as a word separator in the baseline.

iv) A single word may consist of more than one sub-word or connected component. Because most of the characters are connected from both sides, left and right. However, there are 11 characters which can be connected from one side only; the right. Those characters are ژ، ۋ، ز، ر، د، ا، ە، و، ۇ، ۈ، ۆ

v) It is written from right to left.

vi) It has no equivalent to capital letters.

vii) It has different shapes depending on their position in the word.

iix) Most of the characters may have the same main body with different stress marks or have the same stress marks with a different main body.

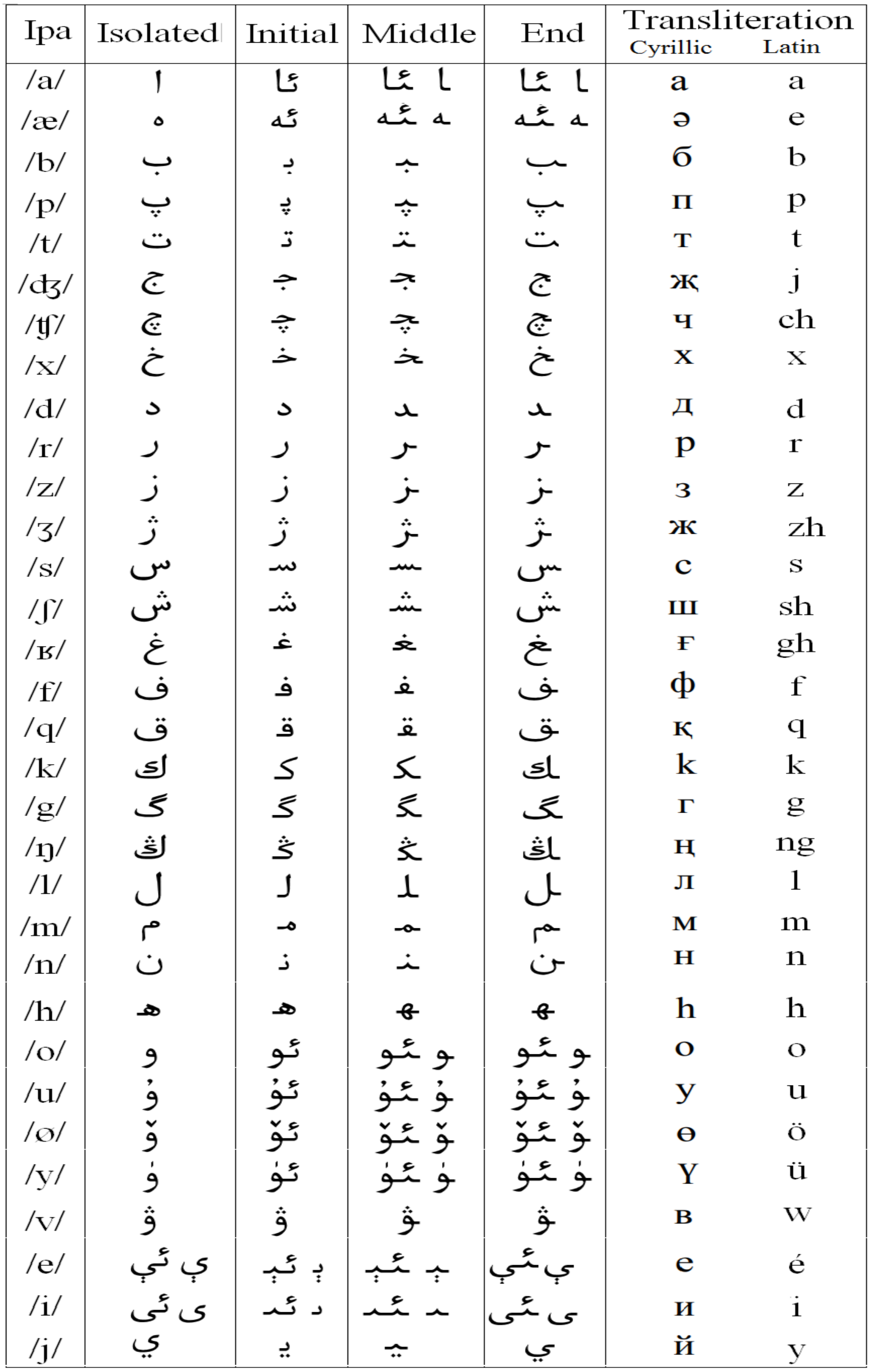
ix) The length or width of a character is variable.

x) Ligature. It occurs when two or more characters overlap vertically and touches. In general, when the one side connected characters, except for د، ا، ە mentioned above, followed by the four characters ج، چ، خ، غ, there is height probability to occur the ligature.

xi) Overlap. It occurs when two or more characters overlap vertically without touching. Similarly, when the one side connected characters, except for the same three characters stressed above, followed by any other characters, the overlaps may occur mostly.

xii) Transliteration, it is concerned with rewriting Uighur words in the Latin alphabet or in the Cyrillic alphabet based on the phonetics of words or characters. It may be useful at times there is no corresponding equivalent for a Uighur word in English or in other languages within the Latin language system. The important utility of the transliteration is that it is convenient for us to perform read/write communication in modern equipment without any extra language input tools. There are special purposed tools to convert the text between the original and its transliteration. Table 2.2 presents the transliteration, IPA (International Phonetic Alphabet) and the shape-changing in four different locations within a word of Uighur characters. The determination and unification of the Uighur alphabet have been proposed in [37].

Table 2.2: The Uighur Alphabet Set. Each character may have up to eight different shapes. The Transliteration of each character to the Latin alphabet and Cyrillic alphabet is illustrated in the right column. The left column is the international phonetics alphabet of the character.



Ipa	Isolated	Initial	Middle	End	Transliteration	
					Cyrillic	Latin
/a/	ا	ئا	ـا ـئا	ـا ـئا	а	a
/æ/	ە	ئە	ـە ـئە	ـە ـئە	ә	e
/b/	ب	بـ	ـبـ	ـب	б	b
/p/	پ	پـ	ـپـ	ـپ	п	p
/t/	ت	تـ	ـتـ	ـت	т	t
/ʤ/	ج	جـ	ـجـ	ج	ж	j
/ʧ/	چ	چـ	ـچـ	چ	ч	ch
/x/	خ	خـ	ـخـ	خ	х	x
/d/	د	د	ـد	ـد	д	d
/r/	ر	ر	ـر	ـر	р	r
/z/	ز	ز	ـز	ـز	з	z
/ʒ/	ژ	ژ	ـژ	ـژ	ж	zh
/s/	س	سـ	ـسـ	ـس	с	s
/ʃ/	ش	شـ	ـشـ	ـش	ш	sh
/ʁ/	غ	غـ	ـغـ	ـغ	ғ	gh
/f/	ف	فـ	ـفـ	ـف	ф	f
/q/	ق	قـ	ـقـ	ـق	қ	q
/k/	ك	كـ	ـكـ	ـك	к	k
/g/	گ	گـ	ـگـ	ـگ	г	g
/ŋ/	ڭ	ڭـ	ـڭـ	ـڭ	ң	ng
/l/	ل	لـ	ـلـ	ـل	л	l
/m/	م	مـ	ـمـ	ـم	м	m
/n/	ن	نـ	ـنـ	ـن	н	n
/h/	ھ	ھـ	ـھـ	ـھ	һ	h
/o/	و	ئو	ـو ـئو	ـو ـئو	о	o
/u/	ۇ	ئۇ	ـۇ ـئۇ	ـۇ ـئۇ	у	u
/ø/	ۆ	ئۆ	ـۆ ـئۆ	ـۆ ـئۆ	ө	ö
/y/	ۈ	ئۈ	ـۈ ـئۈ	ـۈ ـئۈ	Ү	ü
/v/	ۋ	ۋ	ـۋ	ـۋ	в	w
/e/	ې ئې	ېـ ئېـ	ـېـ ـئېـ	ـې ـئې	е	é
/i/	ى ئى	ىـ ئىـ	ـىـ ـئىـ	ـى ـئى	и	i
/j/	ي	يـ	ـيـ	ـي	й	y

The distinct difference between the Uighur and Arabic writing is concluded as following:

i) The character number. Uighur language has 32 basic characters, while the Arabic language has 28 basic characters. Uighur alphabet involves 8 vowels ا، ە، ې، ى، و، ۇ، ۈ، ۆ, but Arabic alphabet just has included three of them.ا، ە، و. Besides, the Uighur alphabet uses the ۋ، ڭ، گ، ژ، پ consonants that the Arabic alphabet does not contain them. At the consonants in Arabic alphabet absence in Uighur alphabet. The Table 2.3 shows the absent and additional characters in the Arabic and Uighur alphabet.

ii) Diacritic marks and zigzags. Diacritics and zigzags are signs that represent short vowels or other sounds in Arabic text. In the Uighur text, we use 8 vowels character to represent short vowels or any sound instead of the diacritic mark and zigzag.

iii) The number of character shapes. Uighur characters have from two to eight different shapes depending on their position within the word, whereas in Arabic each character can contain from one to four shapes according to its location within the word.

iv) The Unicode number in Unicode Standard form. Except for English characters are represented in ASCII form by one byte, the other popular language characters are represented in Unicode form by one byte (width byte) also. The identical character in the Arabic and Uighur language has different Unicode representation codes in the Unicode form. It is one of the crucial reasons why Arabic OCR systems cannot directly apply to recognize the image of Uighur documents.

Table 2.3: The absence and additional characters in the Uighur and Arabic alphabet.

	Uighur	Arabic
Additional Character	ۋ،ڭ،گ،ژ،پ	ط،ظ،ص،ض،ح،ع،ث

2.2. Existing Solutions

Offline typeset Uighur text recognition system with an analytical strategy approach can be divided into the following stages: Image acquisition, Preprocessing, Segmentation, feature extraction, and classification.

2.2.1. Image acquisition

To acquire the text and change it into a digital image via the related input device such as scanner, digital camera and so on.

2.2.2. Preprocessing

To compensate for the poor-quality image by smoothing, diminishing noise and reducing data variation. The smoothing process is made by a mathematical morphology operation: opening and closing operation. The opening and closing operation also use the same basic morphological operations such as dilation and erosion, but in the opposite order [38]. The mathematical morphological operations such as dilation and erosion, but in the opposite order [38]. The mathematical morphology is a tool for extracting image components that are useful in the representation and description of region shape. The language of the mathematical morphology is set theory. The sets in mathematical morphology represent objects in an image [38]. Obtaining the best structuring element to be implemented is the most difficult and important points of those mathematical morphology operations [38]. The structuring element is a small set or a mask shape with a defined origin. The shape would be any shape with any size that digitally representable[38]. The following are definitions and formulas of the erosion, dilation, opening and closing operation.

Erosion is to reduce the number of pixels from the boundary of the object in an image. It is used to remove image boundaries of the region of foreground pixels.

$$A \ominus K = \{x \in Z^2 | x + b \in A \; for \; every \; b \; \in K\} \tag{2.1}$$

Dilation is to add pixels to the boundaries of the object in an image. It is used to bridging gaps

$$.A \oplus K = \{x \in Z^2 | x = a + b \; for \; some \; \; a \in A \; and \; b \in K\} \tag{2.2}$$

Opening is to erode by structuring element then dilates the result by structuring element again. It is used to smooth the country and eliminates protrusion.

$$AoK = \{(A \ominus K) \oplus K\} \tag{2.3}$$

Closing is to dilate and then erode by the same structuring element. It is used to fill gaps in contours [39]

$$A \bullet K = \{(A \oplus K) \ominus K\} \tag{2.4}$$

As to the noise reduction, the noise is unwanted signal and there are two defined type of noise: signal independent noise which adds a random set of grey levels to the pixels in the image and signal-dependent noise in which the value at each point in the image is a function of the grey level [2]. Those noises are diminished by binarization operation.

Binarization or thresholding is a conversion from a grey level image to a bi-level image. The grey level image is also called a grayscale image, is one in which the value of each pixel is a single sample representing only an amount of light, that is, it carries only intensity information. It is composed exclusively of shades of gray. The contrast ranges from black at the weakest intensity to white at the strongest [40]. The bi-level image is also called a binary image or two-level image, is a digital image that has only two possible values for each pixel. Typically, the two colors used for a binary image are black and white. The color used for the object(s) in the image is the foreground color while the rest of the image is the background color [41].

By the Binarization operation, the ancient, old, stained, noised and faded document image could be changed into a clear, clean bi-level text image. The bi-level image contains essential information concerning the position and shape of objects and loses some detailed information.

The crucial problem of binarization is automatic calculating the threshold value for the local and global thresholding. There is a significant amount of research paper about calculating the threshold value, we introduce them from two to three approaches representatively.

The global binarization algorithms are the following:

i) Binarization Based on Classification. This procedure is performed by classifying all pixels into the black and white classes. Choosing the threshold value by minimizing the within-class variance of the threshold black and white pixel, is called Otsu [42][43], by maximizing the between-class variance of the thresholded black and white pixel, is called Reddi [43][44]. The minimization of within-class variance is tantamount to the maximization of between-class variation scatter.

$$T_{otsu} = argmax\left\{\frac{P(T)[1-P(T)]\left[m_f(T)-m_b(T)\right]^2}{P(T){\delta_f}^2(T)+\left[1-P(T){\delta_b}^2(T)\right]}\right\} \tag{2.5}$$

where the $\mathrm{P(T)}$ is the cumulative probability function and it is calculated as $\mathrm{P(T)} = \sum_{\mathrm{g=0}}^{\mathrm{T}} \mathrm{p(g)}$. The $\mathrm{p(g)}$ is the probability mass function of the image. The $\mathrm{m_f(T)}$ is the mean of the foreground $\mathrm{m_f(T)} = \sum_{\mathrm{g=0}}^{\mathrm{T}} \mathrm{gp(g)}$, and the $\mathrm{m_b(T)}$ is the mean of the background $\mathrm{m_b(T)} = \sum_{\mathrm{g=T+1}}^{\mathrm{G}} \mathrm{gp(g)}$. The $\delta_\mathrm{b}(\mathrm{T}), \delta_\mathrm{b}(\mathrm{T})$ are the foreground and background standard deviation. The δ^2 is the variance of the whole image. $\delta_\mathrm{f}^2(\mathrm{T}) = \sum_{\mathrm{g=0}}^{\mathrm{T}}[\mathrm{g} - \mathrm{m_f(T)}]^2\mathrm{p(g)}$, $\delta_\mathrm{b}^2(\mathrm{T}) = \sum_{\mathrm{g=T+1}}^{\mathrm{G}}[\mathrm{g} - \mathrm{m_b(T)}]^2\mathrm{p(g)}$.

ii) Binarization Based on Histogram Analysis. The histogram is smoothed iteratively by using the three-point mean filter until it has only two local maxima, is called Prewitt and Mendelsohn [45]. The threshold is set to the highest gray-level of the histogram that maps at least (100-p) % of the pixels into the background and the others is the foreground, p which is the black pixel percentage is known (set p=5), is called Doyle[46]. Use an approximation function and minimize the sum of squares between a bi-level function and the histogram then obtain the solution by iterative search, is called Ramesh [43][47].

$$T_{rames\ h} = min\left[\sum_{g=0}^{T}[b_1(T) - g]^2 + \sum_{g=T+1}^{G}[b_2(T) - g]^2\right] \quad (2.5)$$

where $b_1(T) = \frac{m_f(T)}{P(T)}$ and $b_2(T) = \frac{m_b(T)}{1-P(T)}$

iii) Binarization Based on Clustering. The gray-level images are clustered in two parts as foreground and background by using the corresponding clustering algorithm, is called K-means method [48].

iv) Binarization Based on Entropy.Consider the input image as two different signal sources which are foreground and background, and getting the threshold value by maximizing the sum of the two-class entropies. The entropic correlation values are defined as:

$$T_{kapur}\ (T) = argmax\{H_f(T) + H_b(T)\} \quad (2.7)$$

where $\mathrm{H_f(T)} = -\sum_{\mathrm{g=0}}^{\mathrm{T}} \frac{\mathrm{P(g)}}{\mathrm{P(T)}} \log \frac{\mathrm{p(g)}}{\mathrm{P(T)}}$ $\mathrm{H_b(T)} = -\sum_{\mathrm{g=T+1}}^{\mathrm{G}} \frac{\mathrm{P(g)}}{\mathrm{P(T)}} \log \frac{\mathrm{p(g)}}{\mathrm{P(T)}}$ is called Kapur [49]. There was an improved version of this approach with a different definition of entropic correlation formula [43][50] is called Yen, as following:

$$H_b(T) = -log\sum_{g=0}^{T}\left[\frac{P(g)}{P(T)}\right]^2 \quad H_f(T) = -log\sum_{g=T+1}^{G}\left[\frac{P(g)}{1-P(T)}\right]^2 \quad (2.8)$$

There is also another approach that selecting the threshold value by minimizing the sum of fuzzy entropies [51] is called Shanbhag.

$$T_{shanb\,hag}(T) = argmin\{H_f(T) - H_b(T)\} \quad (2.9)$$

where $H_f(T) = -\sum_{g=0}^{T}\frac{P(g)}{P(T)} log[\mu_f(g)]$

$$H_b(T) = -\sum_{g=T+1}^{G}\frac{P(g)}{1-P(T)} log[\mu_b(g)]$$

$$\mu_f(T-i) = 0.5 + \frac{p(T) + \cdots + P(T-1-i) + p(T-i)}{2P(T)}$$

$$\mu_b(T+i) = 0.5 + \frac{p(T+1) + \cdots + P(T-1+i) + p(T+i)}{2\big(1-P(T)\big)}$$

v) Binarization Based on Gaussian Distributions. Transforms the binarization problem to a minimum-error Gaussian density fitting problem by fitting of the mixture of Gaussian distributions [52] is called Kitle.

$$T_{kitle} = argmin \left\{ \begin{matrix} P(T)log\sigma_f(T) \\ +[1-P(T)]log\sigma_b(T) - P(T)logP(T) \\ -[1-P(T)]log[1-P(T)] \end{matrix} \right\} \qquad (2.10)$$

where $\delta_f(T)$, $\delta_b(T)$ are foreground and background standard deviations. Considers equal variance Gaussian density functions, and minimizes the total misclassification error via an iterative search, is called Lloyd [66].

$$T_{\text{Lloyd}} = argmin \left\{ \frac{m_f(T) + m_b(T)}{2} + \frac{\sigma^2}{m_f(T) - m_b(T)} log \frac{1-P(T)}{P(T)} \right\} \qquad (2.11)$$

where σ^2 is the variance of the whole image. The Iterative thresholding is one of the first iterative schemes based on two-class Gaussian mixture models. At iteration n, a new threshold T_n is established by using the average of the foreground and background class means. In practice, iterations terminate when the changes$|T_n$-$T_{n+1}|$ become sufficiently small, is called Ridler[54].

$$T_{\text{ridler}} = \lim_{n\to\infty} \frac{m_f(T_n) - m_b(T_n)}{2} \qquad (2.12)$$

where $m_f(T_n) = \sum_{g=0}^{T_n} gp(g)$, $m_b(T_n) = \sum_{g=T_n+1}^{G} gp(g)$

The local binarization algorithms (determine the threshold values based on the local properties of an image, e.g. pixel-by-pixel or region by region) are the following:

i) Binarization Based on Classification. Gray-level of the image feeds the Kohonen SOM neural network classifie to classify the classes of the foreground and background pixel by using a mapping procedure, is called Papamarkos [55].

ii) Binarization Based on Local Variation. Calculates a local threshold for each pixel by using the local mean value and the local standard deviation in the neighborhood of the pixel, is called Niblack [56].

$$T_{\text{niblack}}\ (i,j) = m(i,j) + k.\sigma(i,j) \tag{2.13}$$

where k=-0.2 and local windows size is b=15. The same method with different and more complicate formula is called Sauvola[57].

$$T_{\text{sauvola}}\ (i,j) = m(i,j) + \left\{1 + k.\left[\frac{\sigma(i,j)}{R} - 1\right]\right\} \tag{2.14}$$

where k=0.4 and R=128. This calculates the threshold value by the mean value of the maximum and minimum values within the neighborhood of the pixel. When the difference of the two values is bigger than a threshold the pixel is considered as the foreground, otherwise, it is a pixel of the background and takes a default value, is called Bernsen [58].

$$T_{\text{Bernsen}}\ (i,j) = 0.5\{max_w[l(i+m,j+n)] + min_w[l(i+m,j+n)]\} \tag{2.15}$$

where w=31, provided contrast $C(i,j) = l_{high}(i,j) - l_{low}\,(i,j) \geq 15.$

iii) Binarization Based on Entropy. Consider the joint entropy of two related random variables that the image gray value at a pixel and the average gray value of a neighborhood centered at that pixel, for any threshold pair, we can calculate the cumulative distribution and then define the foreground entropy from the histogram of grey image, is called Abutaleb[59]

.

$$H_{\mathrm{f}} = \sum_{i=1}^{T}\sum_{j=1}^{\bar{T}} \frac{p(g,\bar{g})}{P(T,\bar{T})} log \frac{p(g,\bar{g})}{P(T,\bar{T})}$$

$$H_{\mathrm{b}} = \sum_{i=T+1}^{G}\sum_{j=\bar{T}+1}^{\bar{T}} \frac{p(g,\bar{g})}{1-P(T,\bar{T})} log \frac{p(g,\bar{g})}{1-P(T,\bar{T})} \tag{2.16}$$

where $\bar{g}$ is the average gray value, $p(g,\bar{g})$ is the histogram of gray image, $(T,\bar{T})$ is the threshold pair, P $(T,\bar{T})$ is the cumulative distribution of the threshold pair, H_{f} and H_{b} are foreground and background entropies. The modified version of this expression has been proposed by redefining class entropies and finding the threshold as the value that maximizes the minimum of the foreground and background entropies, is called Brink[60].

$$T_{\mathrm{brink}} = argmin\{H(T)\} \tag{2.17}$$

where

$$H(T) = \sum_{g=0}^{T} p(g)\left[m_f(T)log\frac{m_f(T)}{g} + glog\frac{g}{m_f(T)}\right] +$$

$$\sum_{g=T+1}^{G} p(g)\left[m_b(T)log\frac{m_b(T)}{g} + glog\frac{g}{m_b(T)}\right]$$

iv) Binarization Based on Neighborhood İnformation. The threshold value is calculated by measuring the local contrast of five 3×3 neighborhoods which are organized in a center-surround scheme. The center neighborhood A_center of the pixel is supposed to capture the foreground (background). While the four 3x3 neighborhoods, called in ensemble A_neigh, in diagonal positions to A(center), capture the background [61] is called Palumbo.

$$\mathrm{B(i.j)} = 1 \tag{2.18}$$

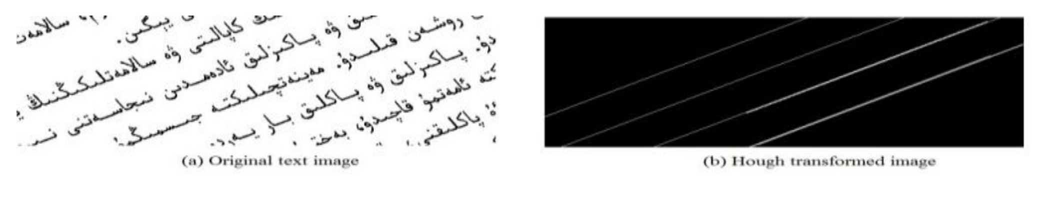
if l (i, j)≤ T_1 or m_{neigh} T_3+T_5 > m_{center} T_4 where T_1=20, T_2=20, T_3=0.85, T_4=1.0, T_5=0, neighborhood size is 3 × 3. Another binarization approaches are given in [43].

Data variation is mainly created by skewing the document image. There are more other data variation reasons but, in this work, we merely concentrate on the skewing image. The solution to the problem is to detect or estimate the orientation angle, which is called skew detection and then rotate up to calculated degrees of the true horizontal in the opposite direction, which is called a skew correction. Skew detection approaches may be categorized into the following groups typically:

i) Skew Detection Based on Projection Profile Analysis. A projection profile is a histogram of the number of black pixel values accumulated along parallel lines taken through the digital image document [62] and the histogram of a digital image represents the frequency of the number of intensity values present in the image [38]. After the horizontal projection profile of text image is calculated at several different angles, determine the angle by maximizing the criterion function which is defined in several forms. The definition of the criterion function: the sum of squared differences in adjacent cells of the projection profile is called Postl [63]. The variance of the number of black pixels in a scan line is called bloomberg1 [64]. The sum of squared values in the projection profile and the sum of squared differences between adjacent values is called bloomberg2 [65]. Define the unit vector as the projection profile on the same direction in any axis and find the unit vector that maximizes the variation. It is called the principal component of the given vector set. Then define the engine vector as the largest absolute engine value of this set. The direction to the unit vector is the exact angle, is called PCA[66].

ii) Skew Detection Based on Nearest Neighbor. After detecting the connected components, build up a histogram from the nearest neighbor pair of those connected components, the peaks in the histogram gives the dominant skew, is called Hashizume[67]. A least-square line fitting process is performed on a subset of plausible neighbors and then build up a histogram from the skew angle associated with the straight line, the peak in the histogram is regarded as the skew angle, is called Chen[68].

iii) Skew Detection Based on Hough line Transformation. The Hough transform is applied to a text image document, the highest number of colinear pixels are the baseline of the text, is called Srihari [69]. Segment the text into blocks or paragraphs then the Hough transform was calculated from the edges of the detected blocks is called ham [70].


(a) Original text image

(b) Hough transformed image

Figure 2.1: The Hough line transformation of the Uighur text image.

iv) Skew Detection Based on Interline Cross-Correlation. Image document is partitioned into several vertical strips, then the angle is determined from the correlation of horizontal projection profiles of the neighboring is called Akiyama [71] and the simplified version that instead of finding the correlation for the entire image, it is calculated over small regions selected randomly and the maximum median of cross-correlation is used as the criterion to obtain the skew[72].

v) Skew Detection Based on Mathematical Operation. Use mathematical morphological operations such as dilation, erosion, opening and closing to smear the text line to solid back bands, then find out the skew of those bands and take the median value as the skew of the entire image is called das [73]. Use structuring element to recursively closing transformant the text image and then binarized it. Finally, we select the dominate direction from the elongated components utilizing an iterative algorithm after applying the recursive opening transformation on the binarized image using the same structuring element, the skew is estimated from selected directions, is called Chen[74].

Given the detected skewness θ, the skew correction is performed by geometric affine transformation by using the Direct Method [75]-[77] that rotating the black pixel by (- θ) shown as Formula (19). The undirected method [78]-[80] is the opposite method of the direct method that

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} cos\ (-\theta) & -\,sin\ (-\theta) \\ sin\ (-\theta) & cos\ (-\theta) \end{pmatrix} \cdot \begin{pmatrix} \acute{x} \\ \acute{y} \end{pmatrix} \qquad (2.19)$$

is computed by applying the inverse rotational matrix to the pixel $(\acute{x},\acute{y})^{t}$ in the goal image shown as formula (2.20). The third method is a counter-oriented method that the de-skewing is applied to the connected components [81].

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \cdot \begin{pmatrix} \acute{x} \\ \acute{y} \end{pmatrix} \tag{2.20}$$

2.2.3. Segmentation

Segmentation is the extracting of the text blocks or columns from document image and decomposing it into text lines, words, sub-words and characters. The segmentation stage is divided into page segmentation, line segmentation, word segmentation, sub-word segmentation and character segmentation substages.

The page segmentation is the process of partitioning images into homogeneous zones, each consisting of only one physical layout structure (text, graphics, pictures, tables) [82]. It is a fundamental problem in computer vision and image processing. The methods of page segmentation can be categorized into foreground analysis, background analysis and hybrid ones from the viewpoint of objects to be analyzed [83]. But the computer vision literature has emphasized three strategies for segmentation, top-down (recursive splitting with backtracking) and bottom-up (iterated greedy merging), and hybrids of these such as split-and-merge [84].

i) The foreground analysis is to form document components (connected components, words, text lines and so on) recursively by collecting black pixels. The representatives of this category include Docstrum, Minimal Spanning Tree and Document Representation. The docstrum or document spectrum is merging the components based on the nearest neighbor-based approach. This algorithm heavily depends on the thresholds that set for a document [85]. The minimal spanning tree considers the whole page as a multi-level graph and the nodes are the connected components, the edges are labeled by the suitable distance between nodes. The distance is defined by several heuristics and the result of it assigned to the nodes. Thus, the words, lines and blocks are detected by merging components iteratively considering the shortest edge in MST at first, second and third layers

[86]. The document representation is splitting a page into columns of text, drawings, images, tables regions, rules then implement region identifying [87].

ii) The background analysis is to segment the page by using straight or curved lines of white pixels. These methods include X-Y Cut, Maximal Empty Rectangles, Whitespace Thinning and Voronoi Diagram. The X-Y cut employ a x-y tree representation of page layout, each node in the tree represents a rectangle in the page and the root node of an x-y tree is the bounding rectangle of the full page. The children of a node are obtained by splitting the rectangle of the parent node horizontally or vertically in the successive level in the tree [88]. Maximal Empty Rectangles is to detect a set of maximal rectangles that do not contain any foreground pixels after split a document image into smaller regions based on the structure of the background. A region classification algorithm could be performed on them [89],[90]. Whitespace Thinning is representing the document with the chain after performing a thinning algorithm on a document, the chains are analyzed to eliminate the lying between the characters and text lines, and to preserve the chains between certain regions of the same type and of different data type [91],[92]. Voronoi Diagram is to label the connected component after removing the noised connected component by using a threshold and then generate Voronoi Diagram by using a sample points method on its border. Finally, delete the suferfluous Voroni edges to obtain the text zone boundary then removing the noise zone using threshold [93].

iii) The Hybrid methods, which is also called global-to-local strategy method, are used for analyzing both foreground and background regions that based on analysis of the whitespace in the image, then look explicitly for white rectangles and find the gaps between column, by that way solve the some raised flaws in a recursive top-down method. Some of the methods have defined the black field separators (BFS) and white field separators (WFS) then merged the connected components into blocks [94], identifies column gaps then group them into column separators[89][95].At first, it uses bottom-up methods to form an initial data-type hypothesis then applies a top-down manner to impose structure [96].

The text line segmentation is determining text and non-text regions in an unconstrained document image. Text line segmentation techniques could be classified into projection-based methods, Hough transform-based methods, grouping methods, repulsive-attractive network methods, and stochastic methods and smearing methods. Several methods cannot be included in a certain class since they do not share a common guideline.

i) The Projection Based Methods are using projection method with other additional processing such as center of gravity[97], divide text image into vertical strips [98], obtained dangles by Hough transform [99], score the potential segmentation points according to their distance between adjacent potential segmentation points [100] and so on.

ii) The Hough Transform Based Methods are using a hypothesis-validation strategy to generate the best text-line hypothesis in the Hough domain iteratively then they check the validity of the line in the image domain using proximity criteria [101]. Considering the fluctuating lines of handwritten drafts, applying Hough transform to get the initial lines then confirm the segmentation points with several repeating steps[102].

iii) The Grouping Methods are extraction methods by using alignment detection iteratively including direction detection and perceptual grouping [103], reconstruct the path of the lines of text using an approach of gradually constructing line segments until a unique line of text is formed by applying the Potential baseline segments (pBLSs) and Baseline segments (BLSs) definition [104].

iv) The Repulsive-Attractive Network Method is detecting the text line by using an energy minimizing dynamic system that interacts with the textual image by attractive and repulsive forces. Thus, the baselines are constructed by scanning the image from top to bottom ordered by grouping neighboring pixels [105].

v) The Stochastic Methods first divide the text image into little $m \times n$ grids and then construct a directed graph then calculate the optimal path based on a probabilistic Viterbi algorithm with Hidden Markov Models (HMMs) [106].

vi) The Smearing Methods are using the Run-Length Smoothing Algorithm to extract text lines [107]. For example, with C = 4 the sequence x is mapped into y as follow:

x :00010000010100001000000011000

y:11110000011111111000000011111

and the different version with dilation to the gradient image along the horizontal direction [108].

After text line segmentation, the next step is the problem of segmenting a line into words. Obtaining the word and sub-word space difference threshold value is called word segmentation.

The calculation of the threshold is conducted in two kinds of approaches which published so far. the one is by using Bayesian minimum classification error criteria:

$$d_o = argmin\big(error(d)\big)$$
$$error(d) = \int_d^{\infty} p_{sw}(d)dx \; + \int_0^{d} p_w(d)dx \qquad (2.21)$$

where $\mathrm{p_w}$(x) and p_{sw}(x) are the probabilities that represent a separation of word and sub-word respectively. These are obtained by manually analyzing the histogram of over a hundred lines [109]. Another is the value of the threshold is taken as half of the text line height [110].

The Character Segmentation is finding the segmentation point on the connected script text. Character segmentation techniques can be divided into histogram or projection and baseline-based methods, contour tracing-based methods, thinning based methods, neural network-based methods, graph theory-based methods, morphological operators-based methods, HMMs based methods, transformation-based methods, strokes, segments and tokens based methods, holistic approach based methods.

i) The Histogram or Projection and Baseline Based Methods are simplifying the problem of the character segmentation into a 1 D system instead of a 2 D system. Some of the methods are: the segmentation points are the least sum of the average summation at the baseline [35],[111],[112], a selected constant threshold value [113], calculate the points by 2 steps such as topological and semi topological segmentation[4], calculate them from vertical projection of the middle zone instead of the entire word with fixed given threshold value and the over-segmentation problem for some characters such as **س،ش** was solved in the recognition phase [5],[114], takes the middle point of the constant amplitude in the profile [3] and with applying some preprocessing techniques such as skeletonization of text before employing the projection profile [115].

ii) The Contour Tracing Based Methods are getting the boundary pixels, reducing the data to be processed and connecting the segmentation stage with the recognition stage [116], solve the overlapping lower or upper strokes in the handwritten text [117].

iii) The Thinning Based Methods extract skeleton of an object that provides its essential information as a single line by using thinning algorithm and detect the segmentation points from the skeleton of the image by using a 3x3 window to identify potential points for segmentation [118] and determining the segmentation point according to the changing direction form candidate start and end points then solve the over segmenting problem[119].

iv) Neural networks-based methods verify the valid segmentation points by using a trained neural network. One of them uses feed-forward multilayer neural networks [120].

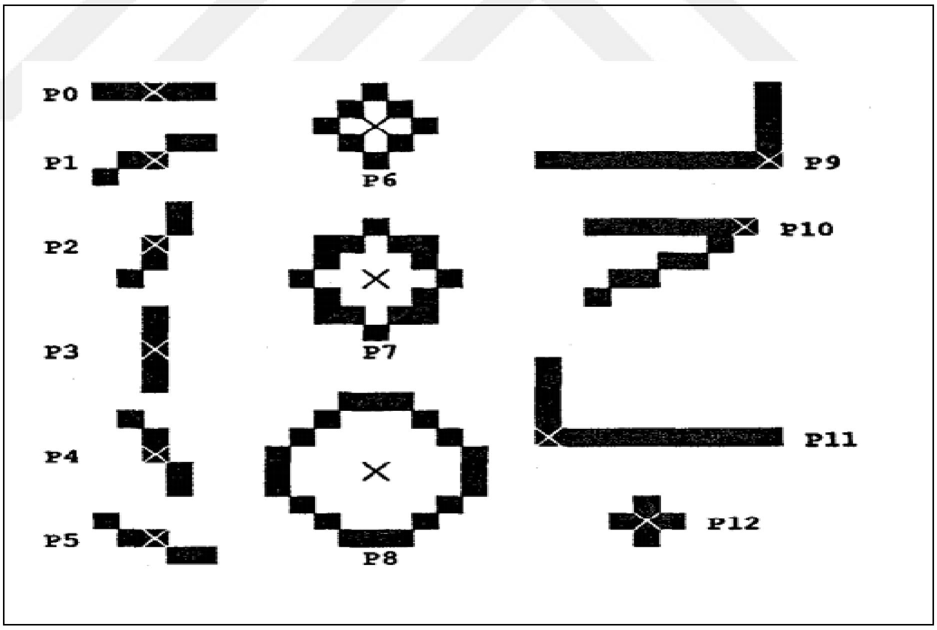


Figure 2.2: The shape primitives (structuring elements) used by the system. Each square corresponds to an image pixel. The origin of each shape is indicated by a cross. (Source: Al-Badr, Badr and Haralick, Robert M)

v) The Graph Theory based Methods obtains segmented characters or the structural feature of the sub-word units by representing the cursive text with some kind of graph representations. For example, LAG (line adjacency graph) which represents the sub-word in the text [121] and Modified MCR (Minimum Covering Run) in Document Parsing including word to character segmentation [122].

vi) The Morphological Operators Based Methods are using morphological operators to simplify the structure of image for purpose of character segmentation by detecting a predefined set of shape primitives on the isolated image of the word [123] shown as Figure 2.2 and by founding the common primitive patterns after thinning different images of a single character then use the Hit-or-miss operator to determine which patterns exist or do not in the input images [124]. The hit-or-miss operator is a morphological operation that performs the task of locating points inside an object with certain (local) geometric properties and is defined by

$$X \otimes (\mathrm{A}, \mathrm{B}) = (\mathrm{X} \ominus \mathrm{A}) \cap (\mathrm{X}^{\mathrm{c}} \ominus \mathrm{A}) \tag{2.22}$$

where $\mathrm{X} \otimes (\mathrm{A}, \mathrm{B})$ is the hit-or-miss-transformed set of X by (A, B), $(\mathrm{X} \ominus \mathrm{A})$ and $(\mathrm{X}^{\mathrm{c}} \ominus \mathrm{A})$ is the erosion of X by A and the erosion of X^{c} by A [125].

vii) The HMMs Based Methods are constructing different numbers of HMM models for each character or ligature for definite fonts and perform the segmentation process by using the models. They use a sliding window to scan each line and extract features to apply to the HMM models [126] and the number of the features is constant 24, the width of the window is fixed, height is equal to line-height [127].

iix)The Transformation Based Methods are using some transformation method to segment character or extract features from Arabic scripts by applying using Hough transform with two versions, Standard Hough Transform (SHT) and Generalized Hough Transform (GHT)[128] by applying the wavelet transform that detecting the underlying horizontal edges and baseline [129].

ix) The Strokes, Segments and Tokens Based Methods Separate horizontally overlapping Arabic word or subword to character fragments and combine them to confirm to be recognized by a feedback loop. They use structural properties, connectivity points and convex dominant points of the Arabic words then extract the features and obtained its concentrated chain code sequence to fed into the string matcher with four databases [130], define a stroke as a continuous curve and feature points consist of cross, branch points, and line ends, Finally, they extract them by finding out its start point and endpoint [29].

x) The Holistic Approach Based Methods which are called segmentation-free segmentation perform the task of segmentation and recognition simultaneously. It is a strategy of the word level Arabic recognition (WLAR) that characterize an Arabic word image with a unique vector (or set of vectors) of morphological features [131], extracted significant geometric features such as lines, arcs, and corners from the input word image, then they used character model library to vote the features[132], etc.

2.2.4. Feature Extraction

The feature is a measurement made on a glyph and combining it into a vector [2]. It contains discriminating information that can distinguish one object from another, and its value set should be small enough to efficiently discriminate among patterns of different classes [133]. Feature types can be classified into three categories: structural features, statistical features, and global transformation.

i) The Structural Features are describing geometrical and topological characteristics of a pattern by representing its global and local properties [134]-[136]. For Arabic and Uighur character, the strokes, bays in various directions, endpoints, the intersection of line segments, loops, stroke positions relative to the baseline, dots and their positions relative to the baseline are considered as structural features[137]-[140]. The stroke is primitive elements make up a character. The bay is the sequence of directions of image tracing. The zigzag is a feature of Arabic character but not of Uighur character.

ii) The statistical features are describing the characteristic measurements of a pattern, for example, zoning which characterize the density distribution of image pixel[141],[142], characteristic loci which calculates the white and black segments then gets length of each segments [143], the ratio of pixel distribution [144],[145], moments which are statistical properties about the intensity of the pixels in image [12],[31],[146],[147], crossing which counts the number of transitions from background to foreground pixels along vertical and horizontal lines through the character image, and distances which calculate the distances of the first image pixel detected from the upper and lower boundaries of the image along the horizontal lines [133].

iii) The Global transformation transforms the pixel representation to a more compact encoding known as series expansion, thus reduces the dimensionality of the feature vector. Common transform and series expansion features are following projection transform which converts a character image of order $M \times N$ into a projection vector of order $M - N$ [62],[112],[148], Fourier Transforms which represent the image shape in a frequency domain that the lower frequency regions contain information about the general features of the shape and the higher frequency regions contain information about finer details of the shape[38],[149]. The Fast Fourier transform is a practical way to compute the Fourier transform of a function [149]. Hough Transform which detects straight lines in a raster image [150], Gabor Transform which is a version of windowed Fourier transform [133], Wavelets Transform which represent the signal at different levels of resolution [151], KarhunenLoeve Expansion which reduce the dimension of the feature set by generating new features from original features[133], Chain-code transformation which represents the thinning or boundary pixels of the character using Freeman code [2],[24],[133],[152],[153].

Feature extraction is capturing and preserving the essential characteristics of the character or the word by filtering out all attributes which make a character or word in one font different from the same character or word in another [2].

2.2.5. Classification and Recognition

Classification is a supervised learning approach in which the computer program learns from the input data given to it and then uses this learning to classify new observations [154]. Historically, classification followed three main paradigms: syntactic (or structural), statistical(or decision-theoretic) and neural network classification paradigms [155] and there are some simpler method that does not fall under above paradigms, for example, dictionary lookup [156],[157], rule-based classification [30] and handcrafted tree classifiers [158]. In this thesis, we focused on the neural network classification paradigms. Neural networks, which is also called artificial neural networks (ANN), is a common, powerful machine learning technique worked by using interconnected networks or simple (and typically) non-linear units

[159]. The machine learning is a software system or mathematical model that uses data to makes predictions without being explicitly programmed[160]. Neural networks are organized in three layers: input layer, hidden layer and output layer. Artificial neural networks may be characterized according to network topology, characteristics of the artificial neurons and the learning or training algorithm used [154]. While neural networks had been studied in the late 60's, they have fallen from grace due to lack of progress and accurate results. However, with the introduction of heavy-computing capable machinery, they have been revived. Contemporary neural network-based research (also known as Deep-Learning) uses multi-level neuron layers which can have more than 1000+layers. Deep-learning methods are representation-learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transforms the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level [161]. According to the structure that suited to different types of tasks, the deep-learning methods have various types. For example, Convolutional Neural Networks (CNNs) are typically used for computer vision tasks, Recurrent Neural Networks (RNNs) are commonly used for processing language. In this thesis, we employ a CNN which is and a multi-layer feed-forward neural network that extracts properties from the input image data by training with neural network back-propagation algorithm. Our CNN is based on LeNet which is a well-known CNN Architecture.

2.3. The New Methods Used in those Solutions

The image acquisition, preprocessing and page, text line segmentation methods for cursive Arabic script as well as non-cursive Latin script have been studied extensively and all of the methods for non-cursive Latin script might carry over to cursive Arabic script. The word and character segmentation are a problem unique to cursive Arabic script, so we concentrate on the two problems in the segmentation stage.

In the classification stage, there are many conventional and representation machine learning algorithms for Arabic OCR and another Arabic script-based OCR, but until now (oct.2019). The all of the published classification methods for Uighur

OCR have belonged to conventional learning algorithm (in google scholar).

The last word segmentation method reported by JH AlKhateeb et al calculates the word and sub-word threshold value by using the word space and sub-word space probabilities [109] and we proposed an alternative method that uses the minimum value to the frequency distribution of widths of all spaces.

The newest character segmentation approach published by Mousa et al gets the segmentation points by low amplitude region in the vertical projection profiles [3]. But this report has no sufficient detail to remove the over and under segmented points.

The new paper about Arabic OCR using representational learning algorithm is reported by Ahmed et al and uses segmented one character as a unit image [162]. But these kinds of methods used in Uighur language are not reported by now.

2.4. Datasets

As to the Dataset used on experiment and training, there are no any open reported, shared or accepted standard training dataset for Uighur characters so far and we prepared a series version of Uighur characters data set that obtained from scanned book papers, synthesized pdf page images which processed by salt pepper noise algorithm respectively.

Text Line Image dataset is used to evaluate the word segmentation experiment, Sub-word image dataset is for character segmentation experiment and three different training/testing character datasets are used to conduct the classification experiment.

2.5. Performance Measurement

To evaluate the OCR system, we use the converted text of line, word and isolated character correctness. The word segmentation is evaluated by the correctness of coordinate to the segmented point with a definite and suitable delta value. The character segmentation is also used the coordination of the cute points with the same definite and suitable delta value. The accuracy of test data the classification result is the metric of the classification experiments.

3. PROPOSED METHODS

As we already mentioned, character recognition systems may be composed of image acquisition, preprocessing, segmentation, feature extraction and classification stages. In our work, three novel approaches are contributed to the segmentation and classification stages. The feature extraction stage is missed out for the characteristic and advantage of the deep learning method used in the classification stage that the feature extraction is made by function and algorithm automatically instead of doing by hand. The other stages or some substages are just implemented with the published, sophisticated, present techniques.

After the image acquisition, the preprocessing stage consists of the compensation image using opening and closing operation, diminishing noise by using binarization, reducing data variation with skew detection and correction operation. Among them, the binarization operation used the Sauvola method [57], because the published report in [163] said this method is better than others. the skew detection used OpenCV library function and the skew correction applied the present OpenCV library function because the three types of methods mentioned in above the literature review chapter have not been evaluated with publishing reports about the difference and efficiency of them.

The following is the segmentation stage which includes the page segmentation, text line segmentation, word sub-word separation and character segmentation substages. Page segmentation also used the present capsulated bounding box function in the OpenCV library. The text line segmentation used the projection-based method [97].

3.1. Word Segmentation

In the word sub-word separation substage, there are only two methods published [109], [110]. The method in [110] was proposed for handwritten text. We propose a calculation method to get word sub-word separation threshold value that the minimum value to the frequency distribution of width of the total spaces including word space and sub-word space in a line or several lines is considered as the word sub-word separation threshold value.

$$T(s) = argmin\left(\sum_{c=1}^{K} N_c\right) \tag{3.1}$$

where T(s) is word and sub-word differenced threshold value, $c$is column width starting from 1 to K which is a random appropriate integer value larger than any word space distance,N_c is the number of spaces whose width is equal to c. Our experiments show that the distribution vector obtained from the space width in several lines need to normalize and we use the median filter algorithm to smooth it

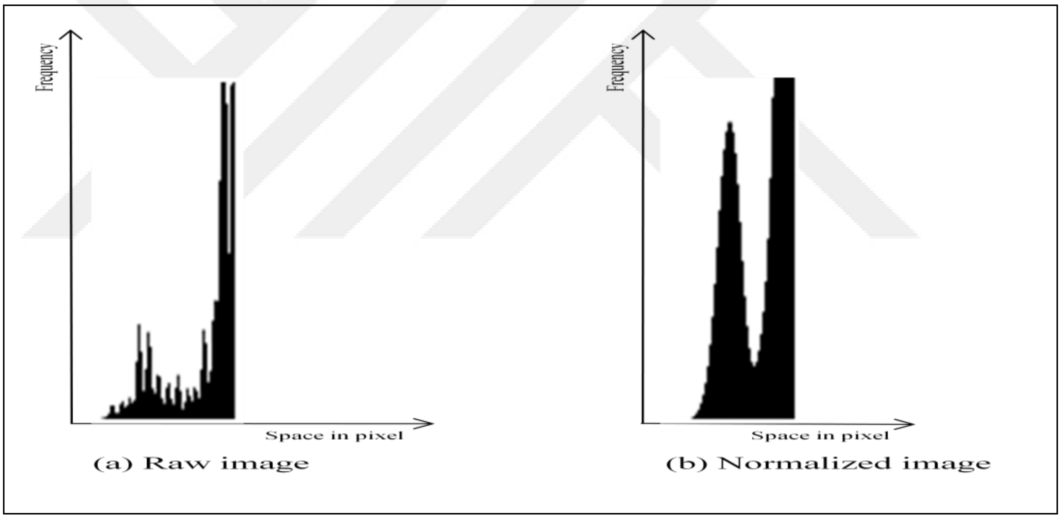


Figure 3.1: The histogram of space between words and sub-words in a typical Uighur text. Space is measured in pixels (horizontal axis). (a) The histogram shows the raw image. (b) Histogram for the normalized image obtained after the normalization process described in the text.

given in Figure 3.1. In this substage, the text line image must shrink one quarter from the top and bottom edges respectively to settle the overlapping probably occurred between characters.

This method is similar to the approach in [109] as it uses the probability of separation of Word P_w and sub-word P_{sw} instead of space width histogram in the text lines.

```
Step-1:  Int function word_subword_threshold (vector<Mat> vctr_textlineimage)
    Return max(histogram of the widthes of all space between word and subword
    in the vctr_textlineimage)

Step-2:  Int function baseline_ycordination(Mat textline_image)
    Return max(horizontal projection profile of the textline_image)

Step-3:  Int function constant_continued_projection_value(vector<Mat> vctr_textline_image)
     Return max(histogram of the values of all constant continued(length>2)
     region in the vertical projection profile of the vctr_textline_image)

Step-4: function character_segmentation (Mat_word_image, İnt_wrd_sbword_threshold,İnt_
            CnstantContndPrjctn_vl, BbsLn_Ycrdntion)
     If (the length of the constant continue region in the vertical projection profile
     of the word_image>2)
          If ((the vertical projection value of this region == CnstantContndPrjctn_vl)
and(the location( the x coordinate the half of this region, the BbsLn_Ycrdntion) is
belong to the black pixel  area) and (the distance between 2 segmentation points>
word_subword_threshold))
              return the x coordinate the half of this region is a segmentation point
```

Figure 3.2: The algorithm of word separation and character segmentation.

3.2. Character Segmentation

As to the character segmentation substage, we propose a projection profile-based method to find the segmentation points in the sub-word image by using constant continued projection value and baseline coordinate which are calculated for each text line image in advance. The constant continued projection value is a special value that the connection points between letters are taken place on the baseline together with the constant continued projection value which is also constant along all text lines even all pages supposing the unified font size. For any Arabic script segmentation, the importance of the constant continued projection value is no less than the importance of the baseline. The main steps of our proposed character segmentation method are given in Figure 3.2.

$$C_{nn\_prj} = argmax\left(\sum_{c=2}^{T} K_c\right) \tag{3.2}$$

In the third step, the calculation approach of constant continued projection value is show as formula (3.2).

where C_{nn_prj} is the constant continued projection value, $c = 2$ is the cumulative number of this continued value that its value is equal to the just previous one in projection profile, in another word, it is the width of constant continues projection value in the profile and start from 2 to T which is a random suitable integer value smaller than the column width of any word, K_c is the value of the constant-continued-projection whose width is equal to c.

In the fourth step, segment the sub-word into characters by using baseline coordinate and constant continued projection value are:

In the projection profile loop, after the encounter the value larger than C_{nn_prj}, if the current index value is equal to the C_{nn_prj} and continue with the same value more than two steps. In addition, if the continuity has ended with a value larger than C_{nn_prj}, then get the half of the continued length as a segmentation x coordinate. If the pixel value is in the x coordinate and the y baseline coordinate is belonging to the black pixel, then the x coordinate is accepted as the segmentation point, go on the loop.

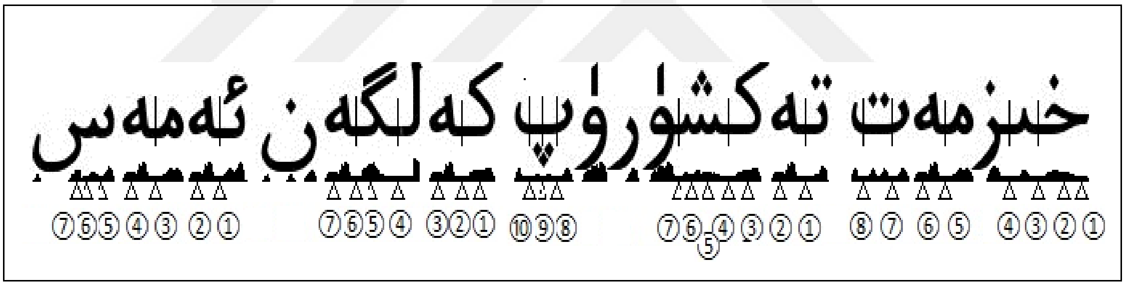


Figure 3.3: Constant continue projection value (constant amplitude) in the vertical projection profile. There are many constant amplitudes with different value, some of them are included in the connected points of characters, whereas some of them are not.

In the Mousa et al's publication [3] also proposed the low variation regions in profile as the segmentation points area, however as the Figure 3.3 showed, there are several low variation (constant amplitude) regions in Arabic scrip projection profile that most of them are not the real segmentation points. In addition, this work didn`t describe the details of how to filter out those unwanted low variance regions that the character connections do not occur in this place. In our work, we observed not only the existence of the constant amplitude but also the calculability of the value to this amplitude that merely related with the segmentation points as long as writing in unified font size. For example, in the Figure 3.3 four words, the 3rd 4th 5th 7th 8th points in the first word, the 1st 4th7th 8th 9th 10th points in the second word, the 2nd 4th 6th points in the third word, 1st 3rd points in the fourth word are have the same projection value that could be directly calculated by formula (3.2). The other indicated low variation regions are not related to the connection points and it needs more operation to remove out those unwanted points based on the approach in [3]. There are five unwanted points (7th 8th points in the first word, 8th 9th 10th points in the second word) calculated in proposed approach, while there are at least eighteen unwanted points ought to be removed (2nd 6th 7th 8th points in the first word, 2nd 3rd 5th 8th 9th 10th points in the second word, 1th 3rd 5th 7th points in the third word, 2nd 4th 7th point in the fourth word) according the description of the approach in [3]. Thus, from the algorithm design, code implementation and computation cost perspective, there are considerable and significant simplification to calculate the segmentation points in Arabic based script which is type set even handwriting provided writing with the same pen width. This means once the value to this constant amplitude is calculated from several lines, the resulting calculated value could be applied to the whole page even to the all pages provided writing in the same font size. Meanwhile, the computation cost also could be reduced.

3.3. Classification Method

The classification stage is the last step in the OCR system, and we use deep

learning technique that takes the raw image data as input and the model hierarchical abstractions in input data with the help of multiple layers. The Convolutional neural network (CNN) is a deep learning architecture that was inspired by information processing in the visua of the human brain. It is a hierarchical neural network and made of one input layer, multi-types of hidden layers and one output layer. The key operation of the CNN is consisting of Input Image, Convolution (Learned), Non-linearity, Spatial pooling, Feature maps as the Figure 3.4 showed.

The LeNet is the classic architecture of CNN that includes an input layer, convolutional layers (2), sub-sampling layers (2), fully connected layers (2) and output layer as Figure 3.5 illustrates. The first layer is accepted the grayscale image with 32×32 sized, the second layer is the first convolutional layer and includes 6 feature maps which are a function that maps a data vector to feature space. Each

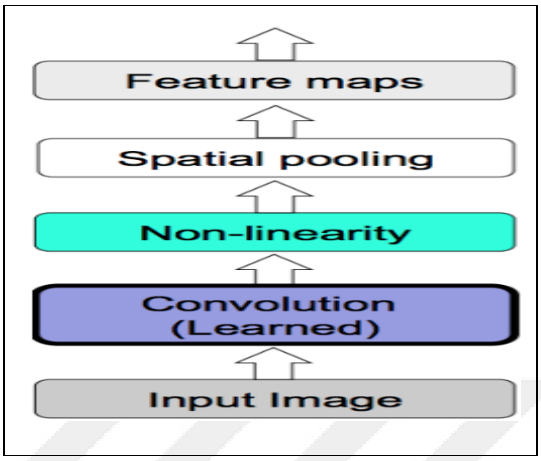


Figure 3.4: The Key Operation of Convolutional neural networks (Source: R. Fergus, Y.LeCun) 5x5 receptive fields at the input layer.

feature map has a 5×5 kernel and the input of each neuron is received by the 5×5 receptive fields at the input layer. The third layer is sub-sampling Layer S2 also has six feature maps and each feature map include a 2×2 kernel. The fourth layer is the second convolutional layer that composes of 16 feature maps and every feature map

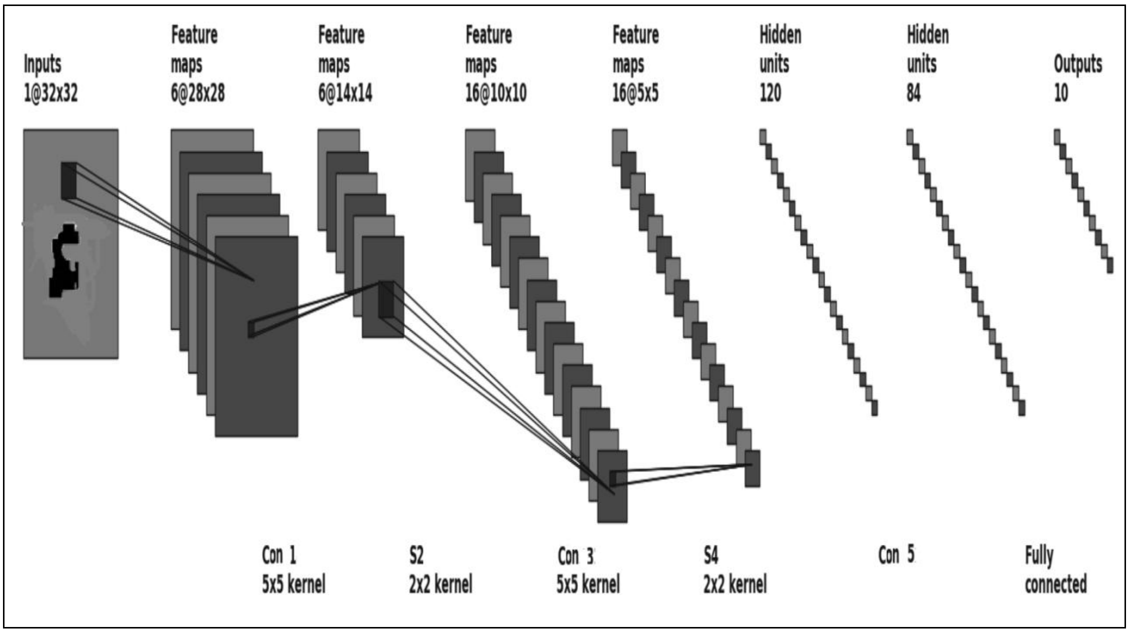


Figure 3.5: Convolutional neural networks LeNet

has a 5 × 5 kernel. The next layer S4 is the second sub-sampling layer with 16 feature maps and each feature map as a 2 × 2 kernel. The sixth layer is the third convolutional layer and involves 120 feature maps; each feature map has a 6 × 6 kernel. The seventh layer is the fully connected layer which selected 84 neurons. The final layer is the output layer. Every CNN layer has the number of connections, parameters and trainable parameters in addition to the feature maps and neurons.

As to be classification unit or input image be classified, we use three consecutive block characters as a classification unit instead of one single character. In the Figure 3.6 showed the three consecutive block characters that each character is cute together with the right and left character after adding a space to the starting and ending of each word. By this manner, the misclassification probability of the character in the middle position would be decreased. In our experiment, we observed decreasing clearly.

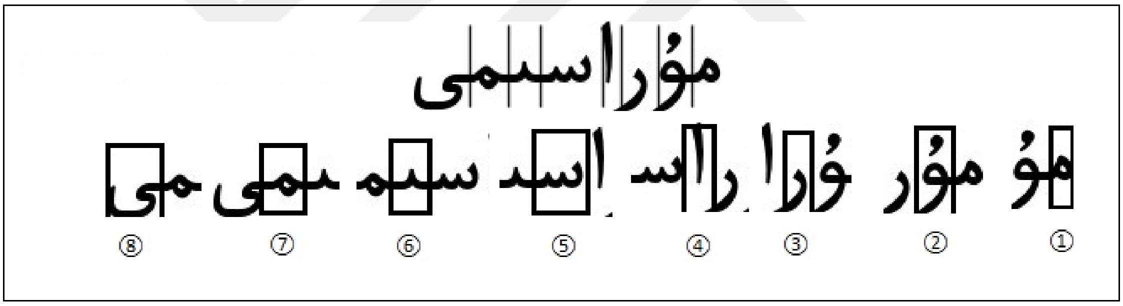


Figure 3.6: Appearance identifies and representations of three consecutive characters series units. The image of each character is represented together with the right and left side characters.

4. EXPERIMENTS AND RESULTS

The methods described in the previous section have been implemented and tested on actual datasets. In this section, we first describe the datasets and then present the details of implementations as well as the results.

4.1. Datasets

In this work, as to be a contribution to the Uighur OCR system we proposed three approaches among the steps or substeps: word sub-word differenced threshold calculation, character segmentation point calculation, and classification. To test the performance of those proposed approaches we use four datasets as following: Text Line Image dataset, Sub-word Image dataset, Three block char dataset and Single char dataset. Thus, test and present the principle advantages of proposed approaches by comparing their results to the corresponding previous approaches that related to our approaches mentioned above. Those datasets, which were collected and generated by us, were fed into the relevant module and the results were quantified in terms of accuracy rates and lost rates.

The Text Line Image dataset contains 1950 text line images collected from scanned book pages. There is a list file that contains the list of the image file name of those text line images and the corresponding x coordinate of the word segmented points in the text line images as to be the standard output results.

The Sub-word Image dataset includes 4379 sub-word images with above 300 dpi, those images are also collected from scanned book pages. There is also a list file that it contains four information: the image file name of those sub-word images, the word and sub-word difference threshold value, the correspondingly coordinate to the baseline of the text line the sub-word be contained and the corresponding to x coordinates of the character segmented points in the sub-word images.

The three-block char dataset consists of 1,210,000 samples with dimensions between $20 \times 60 - 30 \times 90$not binarized and salt-peppered noised images, which were generated and handled by noised procedure, from 10,000 pages Uighur corpus, in 121 classes that each class contains 10,000 consecutive three character images, whose middle one have the representative identity of them shown as figure 11. There are 968,000 training images (80%) and 242,000 test images (20%). The classes are completely mutually exclusive and there is no overlap between them. Those images per class are included in a definite folder that the folder name or directory name which gives the meaningless Unicode number as directory name indicates the label of those images in this folder. The dataset no contains any other file.

The single char dataset also consists of 1,020,000 with 30×90 dimension, 3 channel images, which were collected from more than 50,000 scanned books as well as the ancient, old, stained, faded page documents, in 102 classes, with 10,000 images per class. In the same way, there are 816,000 training images (80%) and 204,000 test images (20%). The classes are also exclusive and no overlap among them. Those images are contained in folders and the names of those folders are named with corresponding Unicode number which indicates the label of those images in them respectively.

The synthesized single char dataset comprises 1,210,000 with dimension 80×80 not binarized and 3 channel images, which were generated and handled by the noised procedure from 10,000 pages Uighur corpus, in 121 classes, with 10,000 single character images per class. There are also 968,000 training images (80%) and 242,000 test images (20%). The classes are exclusive and no overlap among them. Those images are contained in folders and the names of those folders are named with corresponding Unicode number which indicates the label of those images in them respectively.

4.2. Experiments

After reading the image file name from the list in the list file, iteratively get the images and calculate the word sub-word difference threshold value based on formula (3.1), continually split the line into words by utilizing the calculated word sub-word difference threshold value, finally compare the resulted x coordinates to the standard output data reading from the list file. During the experiment based on the proposed approach, among the 11,556 word segmentation points in the 1,950 text line images, correctly calculated the 11,057 points and obtain the 499 wrong points that include the points failed to found the correct position and the unwanted points. The correctness is calculated from the results data with an appropriate delta value that indicates the tolerance distance from the proper position to the point. Our accuracy on this dataset is 95.68%. The suitable delta value is set to the $\frac{1}{10}$ of the height of the text line image.

Similarly, to experiment the calculation of the constant continued projection value or character segmentation point value, perform the same procedure but based on the formula (3.2). Setting the delta value as $\frac{1}{10}$ of the height of the text line image is the best performance during the demonstration. in this experiment, output the 319 error segmented points including both under segmented and over segmented points among the 6,065 segmentation points in the 4,379 sub-words, the correctness is 94.74%.

To classify the segmented character we proposed an approach that uses the three consecutive character as to be input image instead of using straightforwardly the single character image as usual, and to systematically test the impact of the proposed modification on classification performance, we conduct the same experiment with the identical nets on general single character image dataset and obtained the results repeatedly, consequently, we pick the trained CNN model with the lowest validation error, then evaluate and confirm it visually by getting the confusion matrix from the model, finally utilize the models to perform the whole character recognition process and record the word recognition accuracy.

In all experiments, we don`t take it seriously the computation times per epoch include training, validation and testing as well as all data transfers. we also neglect the impact of changing the learning rate after every epoch and experiment with different learning rate initialization constantly thorough the test then repeat it until yielding consistent results, our experiment shows that 0.01 is a suitable learning rate for our datasets and experiment. The loss and accuracy rates of training and validating set for every epoch are shown in Figures 4.1, Figures 4.2, Figures 4.3 and Figures 4.4.

After trained within five epochs, the validating accuracy is 99.33% on the

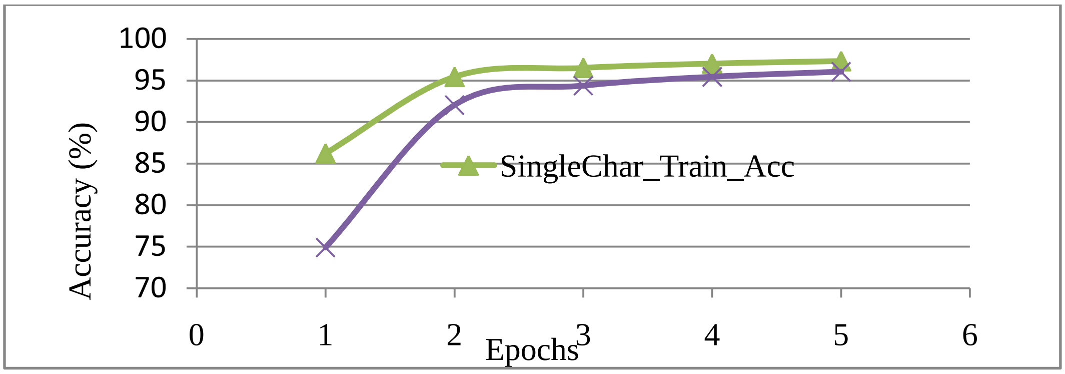


Figure 4.1: The comparison of the training accuracy between single characters and three block characters within the first five epochs.

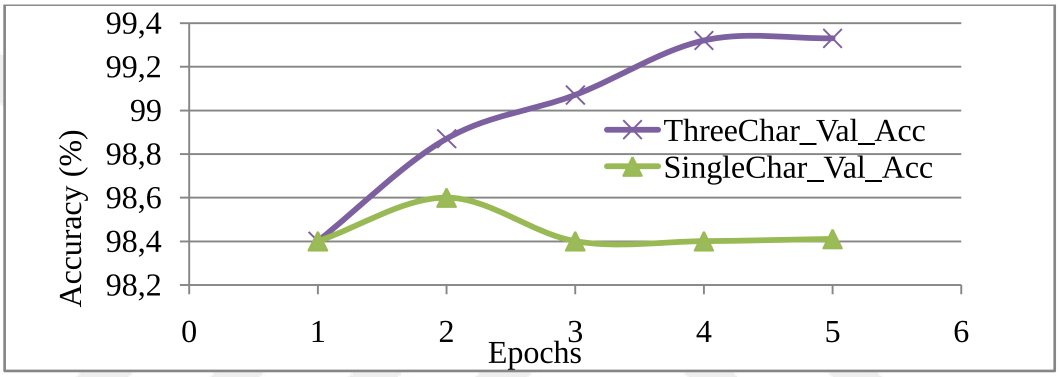


Figure 4.2: The comparison of the validating accuracy between single characters and three block characters within the first five epochs.

three Block Char dataset while it is 98.41% on the Synthesized Single Char dataset and 91.98% on the scanned Single Char dataset. The word recognition accuracies are 88%, 84% and 80% respectively without any NLP post-processing.

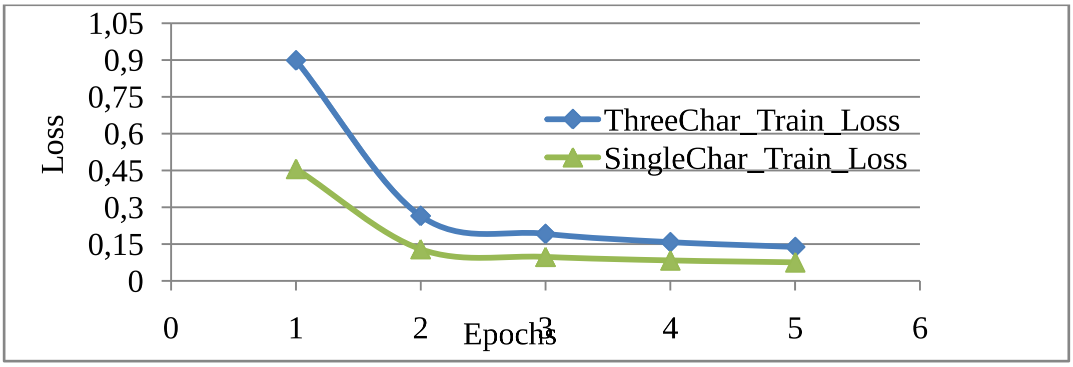


Figure 4.3: The comparison of the training loss between single characters and three block characters within the first five epochs.

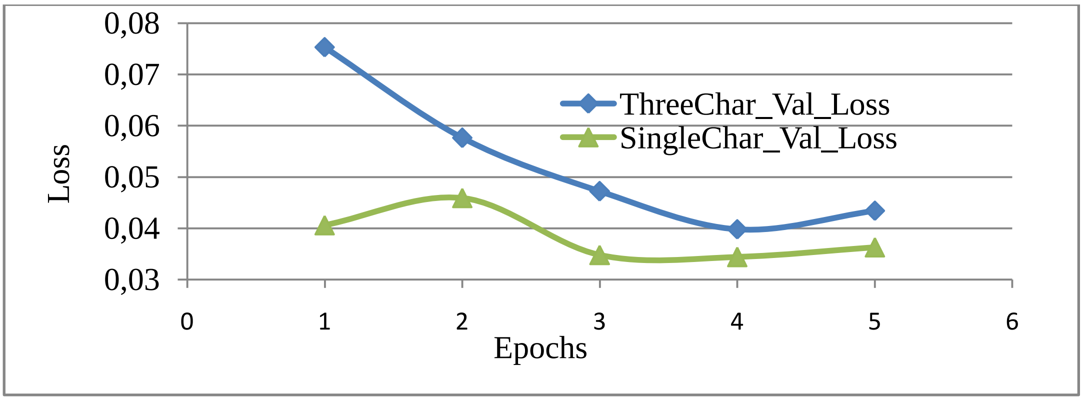


Figure 4.4: The comparison of the validation loss between single characters and three block characters within the first five epochs.

The experiment results of our proposed approaches showed in Table 4.1 and Table 4.2. As to the determination of epoch number what indicates the efficiency and completeness of learning extents and training time, We output a Figure 4.5 may give us a reference or consensus about the appropriate epoch number for such task.

Table 4.1: The performance of the proposed approaches.

Approaches	Performance
Word&Subword Space Difference Threshold Calculation	95.68%
Segmentation Point Calculation	94.74%
Character Image Representation	99.33%

Table 4.2: The overall performance of our system on the scanned text line.

Substages	Performance
Text line to word	95.68%
Word to sub-word	93.74%
Sub-word to character	88.33%
Lenet-5 Classification	80.24%

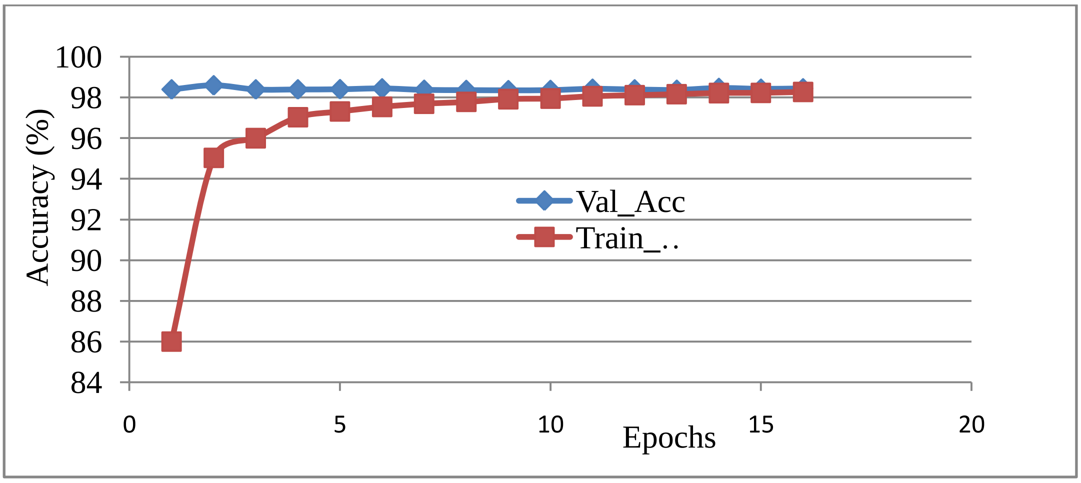


Figure 4.5: Training performance for the three-character method. Training accuracy increases while validation accuracy stays relatively flat. We early stop the training at Epoch 16 and report the results.

5. CONCLUSION

After briefly describing the state of art on Optical Character Recognition methods for the Uighur language, we propose three novel approaches to be used in developing Uighur OCR systems. These methods have been tested and shown to work on realistic datasets which are generated and collected from scanned books or synthesized digitized texts.

First, we introduced a segmentation method that discriminates between word and sub-word. This method has resulted in 95.68% accuracy on our datasets. This method can be improved further if a different normalization on the space width distribution vector is applied and the line text images are filtered by a better thresholding algorithm. The state of the method [109] proposed for handwritten character recognition yields 85% accuracy on their dataset. This method involved more operation that space and time complexity is higher than our method.

Our character segmentation point calculation approach has produced 94.74% accuracy in segmenting the points. Similarly, if the normalization and thresholding algorithm replaced with another algorithm it might have resulted in better segmentation accuracy. The method is easy to implement. In this thesis, we emphasize the importance of the constant continued projection value is no less than the importance of the baseline in Arabic texts. Any Arabic script has a baseline whose location can be calculated as well as there is constant continued projection whose value is calculable and constant in the same font size.

Three-block character image representations also are shown to give better results. Our experiments show 99.3% accuracy compared to 98.4% accuracy of the traditional single character image representation after 5 epochs in training. This is expected as looking around the character would give a better context (like n-gram methods in natural language processing). Our deep learning method is able to make use of this context.

It should be noted that the reported accuracy for our Uighur OCR system looks lower than that of texts in the Latin alphabets. One reason for this is that our method does not employ any NLP processing or word correction procedure during the montage of the words. The results are still promising, and these three approaches can be used in other Arabic script-based OCR systems.

One difficulty in this line of study is the lack of data to train and test Uighur OCR systems. Therefore, unification and extension of the existing data sources would be useful. Another possibility to handle the lack of data is to use transfer learning techniques popular in computer vision and natural language processing fields.

Another avenue of further research is the use of natural language processing (NLP) techniques as constraints in the OCR algorithms. Both segmentation and recognition methods would benefit from the NLP methods such as n-gram word or character filtering.

REFERENCES

[1] Mahmoud S. A., (1994), “Arabic character recognition using Fourier descriptors and character contour encoding”, Pattern Recognition, 27(6), 815–824.

[2] Khorsheed M. S., (2002), “Off-line Arabic character recognition–a review”, Pattern Analysis &Applications, 5(1), 31–45.

[3] Mousa M. A., Sayed M. S., Abdalla M. I., (2013), “Arabic character segmentation using projection-based approach with profile’s amplitude filter”, International IEEE Conference of Information and Communication Technology,122-126, Bandung, Indonesia, 20 – 22 March .

[4] Ymin A., Aoki Y., (1996), “On the segmentation of multi-font printed Uygur scripts”, 13th International IEEE Conference on Pattern Recognition, 215–219, Vienna, Austria, 25-29 Aug.

[5] Nawaz S. N., Sarfraz M., Zidouri A., Al-Khatib W. G.. (2003), “An approach to offline Arabic character recognition using neural networks”, 10th IEEE International Conference on Electronics, Circuits and Systems, 1328–1331, Sharjah, United Arab Emirates, 14-17 Dec.

[6] Zidouri A., (2010), “On multiple typeface Arabic script recognition”, Research Journal of Applied Sciences Engineering and Technology, 2(5), 428–435.

[7] Simayi W., Ibrayim M., Tursun D., Hamdulla A., (2015), “Survey on the features for recognition of on-line handwritten Uyghur characters”, International Journal of Signal Processing, Image Processing and Pattern Recognition, 8(9), 45–58.

[8] Simayi W., Ibrayim M., Tursun D., Hamdulla A., (2016), “A survey on the classifiers in on-line handwritten Uyghur character recognition system”, International Journal of Hybrid Information Technology, 9(3), 189–198.

[9] web 1 (2015), “Lenet-5, convolutional neural networks”, http://yann.lecun. com/exdb/lenet(access date: 30/03/2020).

[10] Ho T. K., (1995), “Random decision forests”, 3rd international IEEE conference on document analysis and recognition, 278–282, Montreal, Quebec, Canada, 14-16 Aug.

[11] Chen T., Guestrin C., (2016), “Xgboost: A scalable tree boosting system”, 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794, August.

[12] Altuwaijri M. M., Bayoumi M. A., (1994), “Arabic text recognition using neural networks”, IEEE International Symposium on Circuits and Systems, 415–418, London, UK, 30 May-2 June.

[13] Cheung A., Bennamoun M., Bergmann N. W., (1998), “A recognition based Arabic optical character recognition system”, IEEE International Conference on Systems, Man, and Cybernetics , 4189–4194, San Diego, CA, USA, 14-14 Oct.

[14] Amin A., (1998), “Off-line Arabic character recognition: the state of the art”, Pattern recognition, 31(5), 517–530.

[15] Amin A., (2000), “Recognition of printed Arabic text based on global features and decision tree learning techniques”, Pattern recognition, 33(8), 1309–1323.

[16] Cheung A., Bennamoun M., Bergmann N. W., (1997), “Implementation of a statistical based Arabic character recognition system”, IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications, 531–534, Brisbane, Queensland, Australia, 4-4 Dec.

[17] McCarthy J., Hayes P. J., (1981), “Some philosophical problems from the standpoint of artificial intelligence”, Readings in artificial intelligence, 431–450.

[18] Duda R. O., Hart P. E., (1973), “Pattern recognition and scene analysis”, Wiley.

[19] Gose E., (1997), “Pattern recognition and image analysis”, 25, 1336-1346.

[20] Byeong-Ho K., (2007), “A review on image and video processing”, International Journal of Multimedia andUbiquitous Engineering, 2(2), 49–64.

[21] Mantas J., (1986), “An overview of character recognition methodologies”, Pattern recognition, 19(6), 425–430.

[22] Tauschek G., (1935), “Reading machine”, US Patent, 2,026,330.

[23] Handel P. W., (1933), “Statistical machine”, US Patent, 1,915,993.

[24] Govindan V., Shivaprasad A., (1990), “Character Recognition—a review”, Pattern Recognition, 23(7), 671–683.

[25] Casey R., Nagy G., (1966), “Recognition of printed Chinese characters”, IEEE Transactions on Electronic Computers,(1), 91–101.

[26] Nazif A., (1975), “A system for the recognition of the printed Arabic characters”, Master’s Thesis, Cairo University.

[27] Ahmed P., Al-Ohali Y., (2000), “Arabic character recognition: Progress and challenges”, Journal of King Saud University-Computer and Information Sciences, 12, 85–116.

[28] Wakahara T., Murase H., Odaka K., (1992), “On-line handwriting recognition”, Proceedings of the IEEE, 80(7), 1181–1194.

[29] Almuallim H., Yamaguchi S., (1987), “A method of recognition of Arabic cursive handwriting”, IEEE Transactions on Pattern Analysis and Machine

Intelligence, (5), 715–722.

[30] Nurul-Ula A., Nouh A., (1988), “Automatic recognition of Arabic characters using logic statements. part i. system description and preprocessing”, J. Eng. Sci. King Saud Univer, 14(2), 343–353.

[31] El-Khaly F., Sid-Ahmed M. A., (1990), “Machine Recognition of optically captured machine printed Arabic text”, Pattern recognition, 23(11), 1207–1214.

[32] Goraine. H, Usher. M, Al-Emami. S. (1992), “Off-line Arabic character recognition”, Computer, 25(7), 71–74.

[33] Plamondon R., Srihari S. N., (2000), “Online and off-line handwriting recognition: a comprehensive survey”, IEEE Transactions on pattern analysis and machine intelligence, 22(1), 63–84.

[34] Amin A., (1987), “Irac: Recognition and understanding systems”, Applied Arabic Linguistic and Signal and Information Processing, 159–170.

[35] Amin A., Mari J. F., (1989), “Machine recognition and correction of printed Arabic text”, IEEE Transactions on systems, man, and cybernetics, 19(5), 1300–1306.

[36] El-Sheikh T. S., Guindi R. M., (1988), “Computer recognition of Arabic cursive scripts”, Pattern Recognition, 21(4), 293–302.

[37] Janbaz W. A., Saleh I., Duval J. R., (2006), “An introduction to Latin-script Uyghur” , Middle East & Central Asia Politics, Economics and Society Conference, 1–15, Salt Lake, USA, 7-9 Sept.

[38] Gonzalez R. C., Woods R. E., (2002), “Digital image processing”, 2nd edition, Prentice Hall.

[39] Matheron G., (1975), “Random sets and integral geometry", Wiley.

[40] Johnson S., (2006),"on digital photography", Inc, O’Reilly Media.

[41] Wu Q., Lu Z., Ji T., (2009), “Protective Relaying of Power Systems Using Mathematical Morphology", Springer, 13–40.

[42] Otsu N., (1979), “A threshold selection method from gray-level histograms”, IEEE transactions on systems, man, and cybernetics, 9(1), 62–66.

[43] Sezgin M., Sankur B., (2004), “Survey over image thresholding techniques and quantitative performance evaluation”, Journal of Electronic imaging, . 13(1), 146–166.

[44] Reddi S., Rudin S., Keshavan H., (1984), “An optimal multiple threshold scheme for image segmentation”, IEEE Transactions on Systems, Man and Cybernetics, (4), 661–665.

[45] Prewitt J. M., Mendelsohn M. L., (1966), “The analysis of cell images”, Annals of the New York Academy of Sciences, 128(3), 1035–1053.

[46] Doyle W., (1961), “Operations useful for similarity-invariant pattern recognition”, Journal of the ACM, 9(2), 259--267.

[47] Ramesh N., Yoo J.H., Sethi I., (1995), “Thresholding based on histogram approximation”, IEE Proceedings-Vision, Image and Signal Processing, 142(5), 271–279.

[48] Jain A.K., Dubes R. C., (1988), “Algorithms for clustering data", Journal of the ACM, Prentice-Hall.

[49] Pun T., (1980), “A new method for grey-level picture thresholding using the entropy of the histogram”, Signal Processing, 2(3), 223–237.

[50] Yen J.C., Chang F.J., Chang S., (1995), “A new criterion for automatic multilevel thresholding”, IEEE Transactions on Image Processing, 4(3), 370–378.

[51] Shanbhag A. G., (1994), “Utilization of information measure as a means of image thresholding”, CVGIP: Graphical Models and Image Processing, 56(5), 414–419.

[52] Kittler J., Illingworth J., (1985), “On threshold selection using clustering criteria”, IEEE transactions on systems, man, and cybernetics, (5), 652–655.

[53] Lloyd D., (1985), “Automatic target classification using moment invariant of image shapes”, IDN AW126, RAE, Farnborough, ReinoUnido.

[54] Ridler T., Calvard S., (1978), “Picture thresholding using an iterative selection method”, IEEE transaction on system, Manefacture, Cybernetc, 8(8), 630–632.

[55] Papamarkos N., Atsalakis A., (2000), “Gray-level reduction using local spatial features”, Computer Vision and Image Understanding, 78(3), 336–350.

[56] Niblack W., (1986), "An introduction to digital image processing", vol 34. Prentice-Hall Englewood Cliffs.

[57] Sauvola J., Pietikäinen M., (2000), “Adaptive document image binarization”, Pattern Recognition, 33(2), 225–236.

[58] Bernsen J., (1986), “Dynamic thresholding of gray-level images”, Proceeding of 8th International Conference on Pattern Recognition, Paris.

[59] Abutaleb A. S., (1989), “Automatic thresholding of gray-level pictures using two-dimensional entropy”, Computer vision, graphics, and image processing, 47(1), 22–32.

[60] Brink A., Pendock N., (1996), “Minimum cross-entropy threshold selection”, Pattern Recognition, 29(1), 179–188.

[61] Palumbo P. W., Swaminathan. P., Srihari S. N., (1986), “Document image binarization: Evaluation of algorithms”, Applications of Digital Image Processing IX, 697, 278–286.

[62] Kasturi R., O’gorman L., Govindaraju V., (2002), “Document image analysis: A primer",Sadhana, 27(1), 3–22.

[63] Postl W., (1988), “Method for automatic correction of character skew in the acquisition of a text original in the form of digital scan results”, US Patent 4,723,297.

[64] Bloomberg D. S., Kopec G. E., (1994), “Method and apparatus for identification of document skew”, US Patent 5,355,420.

[65] Bloomberg D. S., Kopec G. E., Dasari L., (1995), “Measuring document image skew and orientation”, Document Recognition II, 2422, 302–317.

[66] Steinherz T., Intrator N., Rivlin E., (1999), “Skew detection via principal components analysis”, 5th International IEEE Conference on Document Analysis and Recognition, 153–156, Bangalore, India, 22-22 Sept.

[67] Hashizume A., Yeh P.S., Rosenfeld A., (1986), “A method of detecting the orientation of aligned components”, Pattern recognition letters, 4(2), 125–132.

[68] Chen M., Ding X., (1999), “A robust skew detection algorithm for grayscale document image”, 5th International Conference on Document Analysis and Recognition, 617-620, Bangalore, India, 22-22 Sept.

[69] Srihari S. N., Govindaraju V., (1989), “Analysis of textual images using the Hough transform”, Machine Vision and Applications, 2(3), 141–153.

[70] Ham Y. K., Chung H. K., Kim I. K., Park R.H., (1994), “Automated analysis of mixed documents consisting of printed Korean/alphanumeric texts and graphic images”, Optical Engineering, 33(6), 1845–1854.

[71] Akiyama T., Hagita N., (1990), “Automated entry system for printed documents”, Pattern Recognition, 23(11), 1141–1154.

[72] Chaudhuri A., Chaudhuri S., (1997), “Robust detection of skew in document images”, IEEE Transactions on image processing, 6(2), 344–349.

[73] Das A. K., Chanda B., (2001), “A fast algorithm for skew detection of document images using morphology”, International Journal on Document Analysis and Recognition, 4(2), 109–114.

[74] Chen S., Haralick R. M., (1994), “An automatic algorithm for text skew estimation in document images using recursive morphological transforms”, IEEE International Conference On Image Processing, 139–143, Austin, TX, USA, 13-16 Nov.

[75] Yu B., Jain A. K., (1996), “A robust and fast skew detection algorithm for generic documents”, Pattern Recognition, 29(10), 1599–1629.

[76] Chiu L. Y., Tang Y. Y., Suen C. Y., (1995), “Document skew detection based on the fractal and least squares method”, Proceedings of the 3rd International IEEE Conference on Document Analysis and Recognition, 1149–1152, Montreal, Quebec, Canada, 14-16 Aug.

[77] Sun C., Si D., (1997), “Skew and slant correction for document images using gradient direction”, Proceedings of the 4th International IEEE Conference on Document Analysis and Recognition, 142–146, Ulm, Germany, 18-20 Aug.

[78] Hinds S. C., Fisher J. L., D’Amato D. P., (1990), “A document skew detection method using run-length encoding and the hough transform”, 10th International IEEE Conference on Pattern Recognition, 464–468, 18-20 Aug.

[79] Le D. S., Thoma G. R., Wechsler H., (1994), “Automated page orientation and skew angle detection for binary document images”, Pattern Recognition, 27(10), 1325–1344.

[80] Amin A., Fischer S., Parkinson A. F., Shiu R.. (1996), “Comparative study of skew detection algorithms”, Journal of Electronic Imaging, 5(4), 443–452.

[81] Ali H. M. B., (1997), “An object/segment-oriented skew-correction technique for document images”, 4th International Conference on Document Analysis an Recognition, 671-674, Ulm, Germany, 18-20 Aug.

[82] Kaur S., Mann P., Khurana S., (2013), “Page segmentation in ocr system a review”, International Journal of Computer Science and Information Technologies, 4(3), 420–422.

[83] Chen K., Yin F., Liu C.L., (2013), “Hybrid page segmentation with efficient white space rectangles extraction and grouping”, 12th International Conference on Document Analysis and Recognition, 958–962, Washington, DC, USA, 25-28 Aug.

[84] Haralick R. M., Shapiro L. G., (1985), “Image segmentation techniques ", Applications of Artificial Intelligence II, 548, 2–10.

[85] Gorman L. O., (1993), “The document spectrum for page layout analysis”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(11), 1162–1173.

[86] Simon A., Pret J.C., Johnson A. P., (1997), “A fast algorithm for bottom-up document layout analysis”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(3), 273–277.

[87] Jain A. K., Yu B., (1998), “Document representation and its application to page decomposition”, IEEE Transactions on pattern analysis and machine intelligence, 20(3), 294–308.

[88] Nagy G., Seth S., (1984), “Hierarchical representation of optically scanned documents”, Proceeding 7th International Conference on Pattern Recognition, 347-349.

[89] Pavildas T., (1991), “Page segmentation by white streams”, Proceeding of 1st International Conference on Document Analysis and Recognition, 945–953,

[90] Breuel T. M., (2002), “Two geometric algorithms for layout analysis”, International workshop on document analysis systems, Springer, 188–199.

[91] Kise K., Yanagida O., Takamatsu S., (1996), “Page segmentation based on thinning of background”, 13th International Conference on Pattern Recognition , 788–792, Vienna, Austria, 25-29 Aug.

[92] Cattoni R., Coianiz T., Messelodi S., Modena C. M., (1998), “Geometric layout analysis techniques for document image understanding: a review”, ITC-irst Technical Report, 9703(09).

[93] Kise K., Sato A., Iwata M., (1998), “Segmentation of page images using the area voronoi diagram”, Computer Vision and Image Understanding, 70(3), 370–382.

[94] Okamoto M., Takahashi M., (1993), “A hybrid page segmentation method ", Proceedings of 2nd International Conference on Document Analysis and Recognition, 743–746, Tsukuba Science City, Japan, 20-22 Oct.

[95] Haralick R. M., (1994), “Document image understanding: Geometric and logical layout”, IEEE Conference on Computer Vision and Pattern Recognition, 385–390, Seattle, WA, USA, 21-23 June.

[96] Smith R. W., (2009), “Hybrid page layout analysis via tab-stop detection”, 10th International IEEE Conference on Document Analysis and Recognition, 241–245, Barcelona, Spain, 26-29 July.

[97] Marti U.V., Bunke H., (2001), “On the influence of vocabulary size and language models in unconstrained handwritten text recognition”, 6th International IEEE Conference on Document Analysis and Recognition, 260–265, Barcelona, Spain, 26-29 July.

[98] Zahour A., Taconet B., Mercy P., Ramdane S., (2001), “Arabic hand-written text-line extraction”, Sixth International Conference on Document Analysis and Recognition, 281-285, Seattle, WA, USA, 13-13 Sept.

[99] Shapiro V., Gluhchev G., Sgurev V., (1993), “Handwritten document image segmentation and analysis”, Pattern Recognition Letters, 14(1), 71–78.

[100] Antonacopoulos A., Karatzas D.,(2004), “Document image analysis for world war ii personal records**",** First International Workshop on Document Image Analysis for Libraries, 336–341, Palo Alto, CA, USA, 23-24 Jan.

[101] Likforman-Sulem L., Hanimyan A., Faure C., (1995), “A Hough based algorithm for extracting text lines in handwritten documents”, 3rd International Conference on Document Analysis and Recognition, 774–777, Montreal, Quebec, Canada, 14-16 Aug.

[102] Pu Y., Shi Z., (1998), “A natural learning algorithm based on Hough transform for text lines extract in in handwritten documents", Series in Machine Perception and Artificial Intelligence Advances in Handwriting Recognition, 141-150.

[103] Likforman-Sulem L., Faure C., (1994), “Extracting text lines in handwritten documents by perceptual grouping”, Advances in handwriting and drawing: a multidisciplinary approach, 117–135.

[104] Feldbach M., Tonnies K. D., (2001), “Line detection and segmentation in historical church registers”, 6th International Conference on Document Analysis and Recognition, 743–747, Seattle, WA, USA, 13-13 Sept.

[105] Öztop E., Mülayim A. Y., Atalay V., Yarman-Vural F., (1999), “Repulsive attractive network for baseline extraction on document images”, Signal Processing, 75(1), 1–10.

[106] Tseng Y.H., Lee H.J., (1999), “Recognition-based handwritten Chinese character segmentation using a probabilistic Viterbi algorithm”, Pattern Recognition Letters, 20(8), 791–806.

[107] Wong K. Y., Casey R. G., Wahl F. M., (1982), “Document analysis system”, IBM journal of research and development, 26(6), 647–656.

[108] LeBourgeois F., (1997), “Robust multi-font OCR system from gray level images”, 4th International Conference on Document Analysis and Recognition, 1–5, Ulm, Germany, 18-20 Aug.

[109] AlKhateeb J. H., Jiang J., Ren J., Ipson S., (2009), “Component-based segmentation of words from handwritten Arabic text”, International Journal of Computer Systems Science and Engineering, 5(1), 54-58.

[110] Alipour M. M., (2013), “A new approach to segmentation of Persian cursive script based on the adjustment the fragments”, International Journal of Computer Applications, 64(11), 21-26.

[111] Amin A., (1988), “Ocr of Arabic texts”, Pattern Recognition, Springer, 301, 616–625.

[112] Amin A., Al-Fedaghi S., (1991), “Machine recognition of printed Arabic text utilizing natural language morphology”, International journal of Man-Machine Studies, 35(6), 769–788.

[113] Abdelazim H., Hashish M., (1988), “Arabic reading machine”, Proceeding in 10th Saudi National Computer Conference. 733–743.

[114] Sarfraz M., Nawaz S. N., Al-Khuraidly A., (2003), “Offline Arabic text recognition system”, International IEEE Conference on Geometric Modeling and Graphics, 30–35, London, UK, England, 16-18 July.

[115] Zidouri A., Sarfraz M., Shahab S., Jafri S., (2005), “Adaptive dissection based subword segmentation of printed Arabic text”, 9th International IEEE Conference on Information Visualisation ,239–243, London,UK, 6-8 July

[116] Bushofa B., Spann M., (1997), “Segmentation of Arabic characters using their contour information”, 13th International IEEE Conference on Digital Signal Processing, 683–686, Santorini, Greece, 2-4 July.

[117] Romeo-Pakker K., Miled H., Lecourtier Y., (1995), “A new approach for Latin/Arabic character segmentation”, 3rd International Conference on Document Analysis and Recognition, 874–877, Montreal, Quebec, Canada, 14-16 Aug.

[118] Al-Sadoun H. B., Amin. A., (1995), “A new structural technique for recognizing printed Arabic text”, International journal of pattern recognition and artificial intelligence, 9(01), 101–125.

[119] Jambi K., (1991), “Design and implementation of a system for recognizing Arabic handwritten words with learning ability”, Illinois Institute of Technology.

[120] Hamid A., Haraty R., (2001), “A neuro-heuristic approach for segmenting handwritten Arabic text”, ACS/IEEE International IEEE Conference on Computer Systems and Applications, 110–113, Beirut, Lebanon, 25-29 June

[121] Elgammal A. M., Ismail M. A., (2001), “A graph-based segmentation and feature extraction framework for arabic text recognition”, 6th International IEEE Conference on Document Analysis and Recognition, 622–626, Seattle, WA, USA, 13-13 Sept .

[122] Zidouri A., (2004), “Oran: a basis for an Arabicocr system”, International Symposium on Intelligent Multimedia, Video and Speech Processing, 703–706, Hong Kong, China, 20-22 Oct.

[123] Al-Badr B., Haralick R. M., (1995), “Segmentation-free word recognition with application to Arabic”, 3rd International IEEE Conference on Document Analysis and Recognition, 355–359, Montreal, Quebec, Canada, 14-16 Aug.

[124] Timsari B., Fahimi H., (1996), “Morphological approach to character recognition in machine-printed Persian words”, Document Recognition III, 2660, 184–192.

[125] Serra J., (1983), "Image analysis and mathematical morphology", Inc, Academic Press.

[126] Gouda A. M., Rashwan M., (2004), “Segmentation of connected Arabic characters using hidden markov models”, International IEEE Conference on

Computational Intelligence for Measurement Systems and Applications, 115–119, Boston, MA, USA, 14-16 July.

[127] El-Hajj R., Likforman-Sulem L., Mokbel C., (2005), “Arabic handwriting recognition using baseline dependent features and hidden markov modeling”, 8th International IEEE Conference on Document Analysis and Recognition, 893–897, Seoul, South Korea, 31 Aug.-1 Sept.

[128] Touj S., Amara N. B., Amiri H., (2007), “Two approaches for Arabic script recognition-based segmentation using the hough transform”, 9th International IEEE conference on Document Analysis and Recognition, 654– 658, Parana, Brazil, 23-26 Sept.

[129] Broumandnia A., Shanbehzadeh J., Nourani M., (2007), “Segmentation of printed Farsi/Arabic words”, IEEE/ACS International Conference on Computer Systems and Applications, 761–766, Amman, Jordan, 13-16 May.

[130] Cheung A., Bennamoun M., Bergmann N. W., (2001), “An Arabic optical character recognition system using recognition-based segmentation”, Pattern recognition, 34(2), 215–233.

[131] Erlandson E. J., Trenkle J. M., Vogt R. C., (1996), “Word-level recognition of multi-font Arabic text using a feature vector matching approach”, Document Recognition III, 2660, 63–71.

[132] Chen C.H., DeCurtins J. L., (1993), “Word recognition in a segmentation free approach to OCR”, 2nd International Conference on Document Analysis and Recognition, 573–576, Tsukuba Science City, Japan, 20-22 Oct

[133] Kumar G., Bhatia P. K., (2014), “A detailed review of feature extraction in image processing systems”, 4th international conference on Advanced computing and communication technologies, 5–12, Rohtak, India.

[134] Rafael C., Gonzalez R. E., Woods., (1992), "Digital Image Processing", Addison Wesley Longman.

[135] Parker J. R., (2010), "Algorithms for image processing and computer vision", John Wiley & Sons.

[136] Simon J.C., (1992), "Off-line cursive word recognition", Proceedings of the IEEE, 80(7), 1150–1161.

[137] Goraine H., Usher M., (1994), “Printed Arabic text recognition”, 4th International IEEE Conference and Exhibition on Multi-Lingual Computing.

[138] Khorsheed M. S., Clocksin W. F., (1999), “Structural features of cursive Arabic script”, BMVC, 1–10.

[139] Amin A., (1998), “Recognition of printed Arabic text using machine learning " ,Document Recognition V, 3305, 62–72.

[140] Amin A., Al-Sadoun H. B., (1994), “Hand printed Arabic character recognition system”, 12th IAPR International Conference on Pattern Recognition, 536–539, Jerusalem, Israel, 9-13 Oct.

[141] Bazzi I., Schwartz R., Makhoul J., (1999), “An omni-font open-vocabulary OCR system for English and Arabic”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(6), 495–504.

[142] Kondybaev N., (1994), “Maximum entropy approach in automatic classification of symbolic images”, Proceeding in 4th International Conference and Exhibition on Multi-Lingual Computing, Cambridge University Press.

[143] Abdelazim H. Y., Mousa A., Saleh Y., Hashish M., (1990), “Arabic text recognition using a partial observation approach”, Proceedings of the 12th National Computer Conference, 427–437.

[144] Fehri M., Ahmed M., (1998), “An optical font recognizing method for Arabic texts”, Proceedings of the 6th International Conference and Exhibition on Multi-Lingual Computing, 5, 7.

[145] Bouslama F., (1996), “Neuro-fuzzy techniques in the recognition of written Arabic characters”, Proceedings of IEEE North American Fuzzy Information Processing, 142–146.

[146] Mahmoud S. A., (1995), “Arabic character recognition using Fourier descriptors and character contour encoding”, Pattern recognition, 27(6), 815–824.

[147] Sanossian H. Y., (1996), “An Arabic character recognition system using neural network”, Neural Networks for Signal Processing VI. Proceedings of the 1996 IEEE Signal Processing Society Workshop, 340–348.

[148] Amin A., Masini G., (1986), “Machine recognition of multi-font printed Arabic texts”, Proceeding in 8th International Conference on Pattern Recognition, 392–295.

[149] Khorsheed M. S., Clocksin W. F., (2000), “Spectral features for Arabic word recognition”, 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings, 3574–3577, Istanbul, Turkey, 5-9 June.

[150] Duda R. O., Hart P. E., (1972), “Use of the hough transformation to detect lines and curves in pictures”, Communications of the ACM, 15(1), 11–15.

[151] Tuteur F. B., (1988), “Wavelet transformations in signal detection”, IFAC Proceedings Volumes, 21(9), 1061–1065.

[152] Freeman H., (1961), “On the encoding of arbitrary geometric configurations”, IRE Transactions on Electronic Computers, 2, 260–268.

[153] Impedovo S., Ottaviano L., Occhinegro S., (1991), “Optical character

recognition—a survey”, International Journal of Pattern Recognition and Artificial Intelligence, 5(01,02), 1–24.

[154] Kanimozhiselvi C., Jayaprakash M. D., Kalaivani M. K., (2019), “Grading autism children using machine learning techniques”, International Journal of Applied Engineering Research, 14(5), 1186–1188.

[155] Al-Badr B., Mahmoud S. A., (1995), “Survey and bibliography of Arabic optical text recognition”, Signal Processing, 41(1), 49–77.

[156] Kahan S., Pavlidis T., Baird H. S., (1987), “On the recognition of printed characters of any font and size”, IEEE Transactions on pattern analysis and machine intelligence, (2), 274–288.

[157] El-Dabi S. S., Ramsis R., Kamel A., (1990), “Arabic character recognition system: a statistical approach for recognizing cursive typewritten text”, Pattern recognition, 23(5), 485–495.

[158] El-Sheikh T. S., El-Taweel S., (1990), “Real-time Arabic handwritten character recognition”, Pattern Recognition, 23(12), 1323–1332.

[159] Schalkoff R., (1992), “Pattern Recognition: Statistical, structural and neural approaches", Inc, john wiley& sons.

[160] Samuel A. L., (1959), “Some studies in machine learning using the game of checkers”, IBM Journal of research and development, 3(3), 210–229.

[161] LeCun Y., Bengio Y., Hinton G., (2015), “Deep learning”, Nature, 521(7553), 436-444.

[162] Ahmed S. B., Naz S., Razzak M. I., Yousaf R., (2017), “Deep learning based isolated Arabic scene character recognition”, 1st International IEEE Workshop on Arabic Script Analysis and Recognition, 46–51.

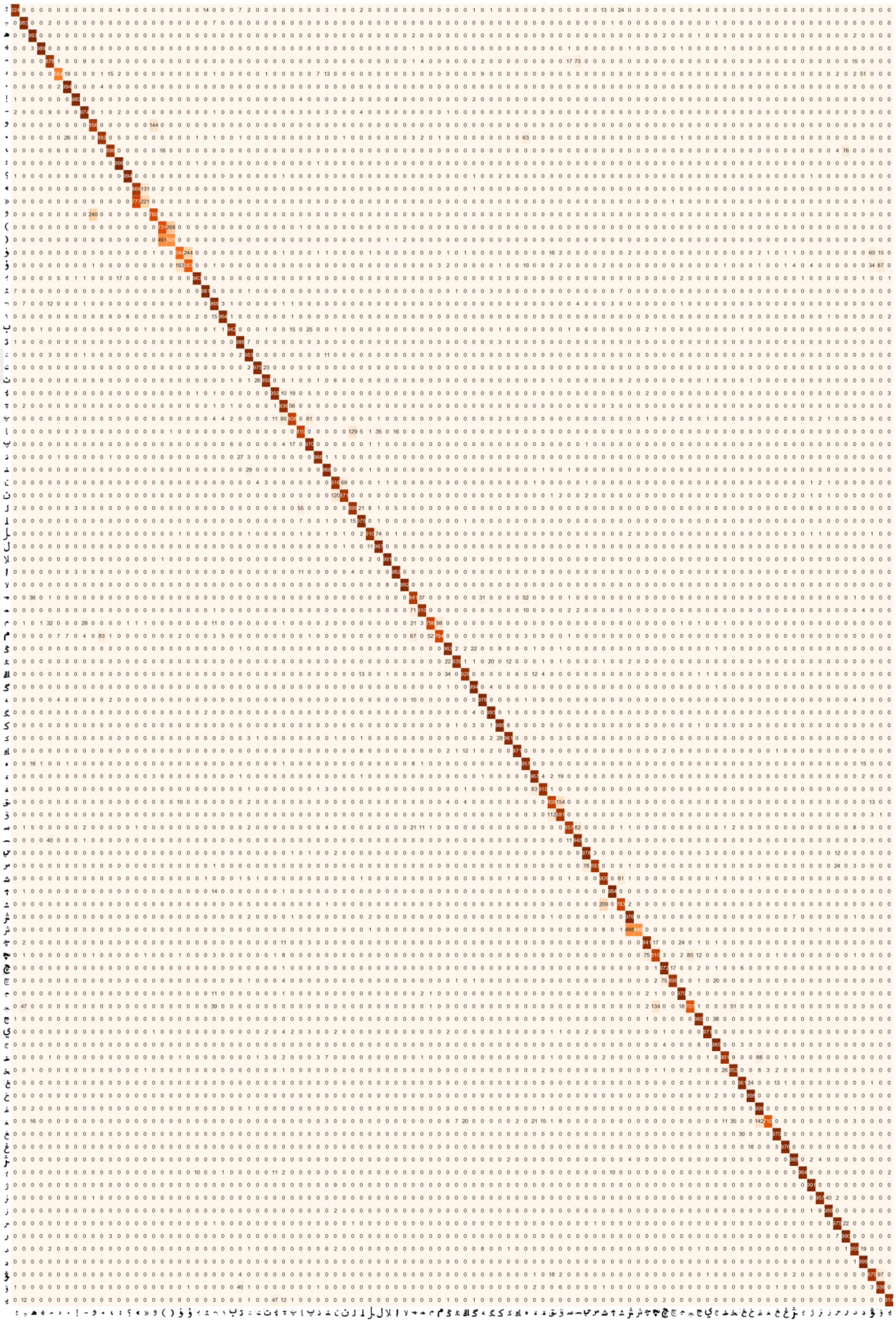
[163] Stathis P., Kavallieratou E., Papamarkos N., (2008), “An evaluation technique for Binarization algorithms”, Journal of Universal Computer Science, 14(18), 3011–3030.

BIOGRAPHY

Memtimin MAHMUT was born in Easten Turkistan in 1979. He took her bachelor's degree from Nanjing University, People Republic of China in 2004. He started her graduate study at Computer Engineering Graduate Program, Institute of Natural and Applied Sciences, Gebze Technical University in 2008. After and before her graduate study he has worked as Computer Engineer in China and Turkey.

APPENDICES

Appendix A: The Confusion Matrix.



The Confusion Matrix of the result of the scanned character trained model. (102 classes and 1000 pictures per class)

Appendix B: The Publicathion During This Work.

Mahmut M., GENÇ Y., (2019), "A Deep-Learning Approach to Optical Character Recognition for Uighur Language", 2019 IEEE International Conference on Advances in Computing, Communication and Control, 1-6, Mumbai, India, 20-21 Dec.