2022-03-25

Designing an Automatic Hand Writer Recognition Model for Amharic Language by Using Convolutional Neural Networks and Support Vector Machine.

Yeshambel, Bekele

BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

FACULTY OF COMPUTING

Designing an Automatic Hand Writer Recognition Model for Amharic Language by Using Convolutional Neural Networks and Support Vector Machine.

BY: Yeshambel Bekele

BAHIR DAR, ETHIOPIA

August 25, 2022

Designing an Automatic Hand Writer Recognition system for Amharic Language by Using Convolutional Neural Networks and Support Vector Machine.

Yeshambel Bekele

A thesis submitted to the school of Graduate Studies of Bahir Dar
Institute of Technology, BDU in partial fulfillment of the requirements for the degree
of
Master of Science in the Computer Science in the Computing Faculty.

Advisor Name: **Abrham Debasu (Ass. Prof)**

BAHIR DAR, ETHIOPIA
August 25, 2022

DECLARATION

This is to certify that the thesis entitled “Designing Automatic hand writer recognition for Amharic language by using Convolutional Neural Network and support vector machine”, Submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science under Faculty of Computing, Bahir Dar Institute of Technology, is a record of original work carried out by me and has never been submitted to this or any other institution to get any other degree or certificates. The assistance and help I received during this investigation have been accordingly acknowledged.

Bahir Dar Institute of Technology
School of Graduate Studies
Faculty of Computing
Request letter for proposal, progress and final thesis presentation

Note.

- The presentation date and time requested by a student is according to the calendar of SRGS. In this case, the student will be notified within one week of receiving the request by the Advisor
- A student will be allowed to take a defense or research study presentation if he/she has completed a number of prior graduation requirements which include (1) **checking the similarity index of the thesis document** and (2) **preparing and submitting research article manuscript for publication to advisors, journals or conferences.**
- This form must be submitted by the student **4 weeks** before the presentation date to his /her advisor.
- Thesis corrections should be implemented by the student based on examination committee's comments and recommendations.

Semester ………………………….…….…... Academic year ……………………..……

Name of student………………………………………………….…………Student ID …………

Degree: ☐ MSc ☐ MEng Program ………………………….…………………………

Email……………………………………………………………Phone Number.…………………

Name of Advisor…………………………………………………Academic rank…………………

Thesis title:

....

I intend to take (select one from the given options) **Examination on** **Time**

☐ Thesis proposal defense	……../……../……..	………………
☐ Thesis progress defense	……../……../……..	………………
☐ Thesis defense	……../……../……..	………………

Supporting documents for thesis defense (select one from the given options)

☐Thesis proposal document

☐Thesis progress report

☐Thesis document

Checked by_____________________________ date______________

_____________________________ ___________________________________

BAHIR DAR UNIVERSITY
BAHIR DAR INSTITUTE OF TECHNOLOGY
SCHOOL OF GRADUATE STUDIES
COMPUTING FACULTY

Approval of thesis for defense result

I hereby confirm that the changes required by the examiners have been carried out and incorporated in the final thesis.

Name of Student **Yeshambel Bekele** Signature ____________ Date 19/12/14

As members of the board of examiners, we examined this thesis entitled "Designing an Automatic Hand Writer Recognition system for Amharic Language by Using Convolutional Neural Networks and Support Vector Machine." by *Yeshambel Bekele*. We hereby certify that the thesis is accepted for fulfilling the requirements for the award of the degree of Masters of science in "Computer Science".

Board of Examiners

Name of Advisor	Signature	Date
Abrham D.		
External Examiner: Dr. Beakal Gizachew		13-Aug-2022
Name	Signature	Date
Name of Internal Examiner	Signature	Date
Gebeyehu B (PhD)		
Name of Chairperson	Signature	Date
Yirga Y. (PhD)		08/24/2022
Name of Chair Holder	Signature	Date
Yirga Y. (PhD)		08/24/2022
Name of Faculty Dean	Signature	Date
Asegahegn E		19-12-2014

Faculty Stamp

Acknowledgements

First of all, I would like to thank the Almighty of GOD, for He is the Source of Wisdom, and the Beginning of Knowledge. I praise and thank Him for His great help throughout this work. I would also thank the Ever-Virgin, St. Marry, Mother of our Lord.

My deepest gratitude is to my advisor, **Abrham Debasu** (Ass.Prof). I am deeply thankful to him for giving his insights and constructive comments. Discussions with him always made me think that things are possible.

Finally, I would like to thank my families, friends, BDU BiT ICT Staff members and Peda Campus ICT workers who in one or another brought me up to success in my academic endeavor. None of this would have been possible without the love and patience of my family and friends. Their support and care helped me to stay focused on my graduate study.

Contents

Abstract

Writer identification is a popular and active research area with numerous applications such as in banking, criminal justice, access control, and detecting the validity of handwritten letters. In this research paper we present a lot of things about designing of an automatic hand writer recognition system for Amharic language by using Convolutional Neural Network and support vector machine. The handwriting styles of an individuals varied from time to time; the reason behind these types of writing styles invariant in each individual are mood, time, space, speed of writing, writing medium and tool. Even though; if it is difficult to identify these writing styles; designing an automatic hand writer system by using Convolutional Neural Network and support vector machine with the integration of applying different features like shape, orientation and angles of character possible to identify the writer of a given document. Writer identification system it is very mandatory especially; In following areas such as forensic expert decision-making systems, network security, digital rights management, biometric authentication in information and document analysis systems, writer identification is critical. According to this study, the process of writer identification, has three basic parts: preprocessing, feature extraction, and identification or classification of the document to the relevant class. In the pre-processing phase; The process of excluding extraneous information in the input data that can negatively affect recognition has been carried out by image resizing, noise reduction, and threshold segmentation. Feature extraction: is a type of dimensionality reduction where a large number of pixels of the image are efficiently represented in such a way that interesting parts of the image are captured effectively. From the total dataset of 2100, 80% (1680) used for training, and 10% (210) used for validation, and 10% (210) also used for testing the model. The testing accuracy of the model is 96%. We used accuracy, precision, recall, f1-score, and confusion matrix to evaluate our model. The model is a combination of CNN-Gabor with SVM Classifier.

Key-Words: SVM, Convolutional Neural networks, Handwritter recognition

List of Abbreviations

BMP	Microsoft Windows Bitmap
CNN	Convolutional Neural Network
COCO	Common Objects in Context
ConvNet	Convolutional Neural Network
CSF	Chine’s Structure Feature
ELM	Extreme Learning Machine
FFT	Fast Fourier Transformation
FNN	Feed Forward Neural Network
GIF	Graphics Interchange Files
GPU	Graphical Processing Unit
JPEG	Joint Photographic Experts Group
KNN	K-Nearest Neighbors
MATLAB	Matrix Laboratory
MINST	Modified National Institute of Standards and Technology
OCR	Optical Character Recognition
Open CV	Open-source Library for Computer Vision
PNG	Portable Network Graphics
ReLu	Rectified Linear Unit
ROI	Region of Interest
SVM	Support Vector Machine
TIF	Tagged Image File Format

List of Figures

List of Tables

CHAPTER ONE

1.1. Introduction

In this chapter, we have discussed the main manifestation and background of handwritter recognition and the creative of data beyond the problem recorded in this work is described briefly. This chapter gives relevant point by raise background information on hand-writter recognition techniques, related documents of the work, the exhibited problems, solutions to be solve the problem, and the expected outcomes are explained greatly.

Writer recognition from handwriting text images has been a popular application and a hot research topic in the fields of computer vision and machine learning in recent years. The process of assigning a handwriting text image to a certain writer based on a set of specified traits, or recognizing a person based on his or her handwriting text images, is known as writer identification(Sulaiman et al., 2019).

It is a common and active research field with various applications areas like, banking, criminal justice, access control, and determining the authenticity of handwritten mails(Daniel Demoze and Eneyew Adugna, 2010).

According to the works of (Maruis Bulacu & Schomaker, 2007) Identification of an individual handwriting using scanned images is a valuable behavioral biometric modality that can be used in forensic and historic record analysis. A writer recognition system identifies the samples that are most similar to the query from a database of known authorship handwritings.

Handwriting recognition poses a variety of problems, including differences in handwriting styles between individuals and variations in handwriting styles within a single individual. Despite the fact that several studies on handwriting recognition have been performed, but achieving an appropriate level of accuracy for handwriting recognition is still a difficult task(Ayalneh, 2020).

The field of handwritten document recognition has sparked a lot of interest in recent years. Because of this there are different works were done in the past and now. But different people's handwritten characters vary in size and shape make it the identification of handwriting is difficult. The identification of individual characters is complicated by the various differences in their writing styles.

The method of recognizing the document's author based on their handwriting is known as writer identification. Recent developments in computational computing, artificial intelligence, data mining, image analysis, pattern recognition, and machine learning have shown that it is possible to automate writer identification characters, complicating the problem even further. The field of recognition encompasses a wide range of topics (Saranya & Vijaya , 2013). In general, this research paper is organized in five chapter that is chapter one it deals about introduction, backgrounds of hand written recognition, objectives of the study, scope and limitation of the research. The second chapter it deals about literature review and related works. Chapter three deals about the methodology, experiments and evaluation results are the concern of chapter four. And finally, we conclude and recommend the future tasks in the area of hand writer recognition under chapter five.

1.2. Background of Hand Writer Recognition

Authentication can be done in a variety of ways to validate a person's identity and it is a great challenge knowing exactly who writes a given document using traditional recognition system. According to the work of (Abushariah et al., 2012) For this challenge, our society had evolved a variety of approaches to authenticated a person. One option is to use possessions, which involves using actual items such as keys, passports, and smart cards. The other method is to use knowledge, which entails memorizing a piece of information that must be kept hidden, similar to a password. Use of the so-called biometrics is the third way for identifying a person.

As described in(Anton et al., 2011),(Daniel Demoze and Eneyew Adugna, 2010),(M. Bulacu, 2007) and (Abushariah et al., 2012) Biometrics is the study of authenticating a person using one or more physiological or behavioral traits that distinguish one person from another. It is derived from two Greek terms, bios, which means "life," and metron, which means "measure." When it comes to biometric identification, when an unknown person needs to be identified, templates with known people are compared. Biometric modalities are divided into two categories: physical and behavioral. Physical biometrics is a method of identifying people by measuring a physical characteristic of the human body (like fingerprint, face, iris, retinal blood vessels, hand geometry, DNA). Individual behavioral attributes such as voice, movement, keystroke dynamics, signature, handwriting, and others, on the other hand, are employed in behavioral biometrics for identification. Because handwriting is the product of a person's action, it is classified as a behavioral biometric feature.

In comparison to physical biometric modalities, behavioral biometrics are less invasive, but the achievable performance is less outstanding due to the high variety of behavior obtained from biometric templates.

As a result, writer identification, the focus of this study, belongs under the domain of behavioral biometrics.

As it is stated in the works of (Daniel Demoze and Eneyew Adugna, 2010),(M. Bulacu, 2007),(M Bulacu & Schomaker, 2006),(Abushariah et al., 2012)Text-dependent and text-independent approaches are the two broad groups for writing identification system. Text-based systems are dependent on text content and only match the same characters, requiring the writer to write the same text as a result. These approaches compare individual characters or words of known text content to a database, and are thus similar to signature verification techniques but more generic in approach than signature identification. As a result, text-dependent systems require the prior segmentation of relevant information by the user or by an automatic segmentation algorithm. In the case of text-independent systems, however, a small amount of handwritten data is required to assure the stability and consistency of data-mined properties. Text independent systems, unlike text dependent systems, which are more constrained and require identical verification text, make no assumptions about the underlying data. They can identify writers for a questioned sample of document without requiring a comparison of the same characters and independent of the text content.

Aside from the preceding explanations of writer identification, there is another way to explain it. It can be either an online or offline writer identification type, depending on the nature of data availability used for the identifying procedure. It is an offline writer identification type when scanned images of handwritten papers (using a scanner, for example) are used. As a result, image processing and pattern recognition techniques such as pre-processing, feature extraction and selection, comparison, and performance evaluation have been used to overcome many sorts of challenges encountered in offline handwriting systems. The second sort of writer identification is the online type. The availability and utilization of temporal (such as velocity, pressure, and stroke order of pen movement) and spatial information of the writing are the foundations of online handwriting systems. Offline handwriting papers do not have this information. Furthermore, when comparing online and offline papers, online handwriting documents contain more information and, as a result, are expected to produce more accurate results. However, both offline and online handwritten document analysis come with their own set of challenges and issues(Li et al., 2009),(Said et al., 2002) and (Abushariah et al., 2012).

The goal of this study is to go text independent rather than text dependent type classification of writer identification systems.

1.3. Statement of Problem

Amharic is Ethiopia's official language. As a result, a variety of handwritten documents, such as meetings, minutes, conferences, action letters to transfer the message from one responsible to others, and other documents, are generated on a daily, weekly, monthly and annual basis in this language script.

Because of the enormous growth in technology and its applications in a wide range of fields, identifying a writer's handwriting has become extremely important. Writer recognition has a broad advantage in different areas such as for crime identifications, forged prevention, digital rights management in the financial sector, forensic expert decision-making and to analysis historic document.

According to (M. Bulacu, 2007)By the help of machine learning; humans are capable of automating pattern recognition tasks such as classification. machine learning can also recognize patterns by passing through learning and classification stages. For large data sets, classifying and categorizing unknown data objects to a corresponding category can take a long time i.e., may be a day, weeks, more or even be impossible for humans. A good approach in this case is to use machine learning algorithms to build a classifier that can categorize data objects into classes with specific output to making a decision.

According to (Marius Bulacu et al., 2003)It is obvious that a handwritten document cannot be rewritten exactly as it was the first time. Changes in handwriting may be caused by the composition of the ink, the writing surface, and the writers' mental state. Because of this; identifying a writer's handwriting by using different parameters like orientation, angle and shape of their writing style is very important to properly identification of handwriting of individuals.

As far as in my searching, there are two research works done towards writer identification system for Ethiopian Amharic Language by [Daniel Demoz, Eneyew Adugna] and Lamesginew Andargie.

According to (Daniel Demoze and Eneyew Adugna, 2010)They propose an Ethiopic writer recognition scheme that is off-line and text-independent by using two characteristics to identify the writer of a given document in their study. The first is a texture level that uses multi-channel Gabor Energy, and the second one is an allograph (character shape) level that uses a connected component contour codebook. In their research, these features are used in a unique way to identify writers of a handwriting document.

They didn't test the impact of combining these features on their writer identification system's results. Furthermore, only the nearest neighbor classification algorithm and the chi-square distance are taken into account. As a result, writer recognition is less accurate than of others researcher’s performance. In the writer identification system, rather than applying a separate effect, merging and combining different features together results a better output.

According to (ALAMIREW, 2015)study he attempts to combine fuse texture and allograph features, as well as use two different classification algorithms. These are Artificial Neural Network and Nearest Neighbor were used to do his research. It is implemented on MATLAB environment and achieve 95.6% of experimental result for 10 hit list size by using Manhattan distance measure on nearest neighbor algorithm is obtained on edge hinge distribution feature. In general, by combining the above listed two algorithms the researcher was achieved 96.4% accuracy for the identification of Amharic writer system. But the researcher doesn’t consider other features like height of writing zones and classification algorithms like SVM.

Based on(M. Bulacu, 2007) Grapheme codebook and edge-hinge features are incorporated, directional, grapheme and run-length information features in (Marius & Lambert , 2006) are combined together for better performance. In (Marius Bulacu et al., 2003) edge-direction distribution, edge-hinge distribution, run-length distribution, autocorrelation, and entropy features are combined and perform better result. In general, based on the above papers result and recommendation combining different features performs better result rather than performing a single feature.

In general, identification of handwritter document has a numerous advantage in different field of study; especially, in courts, banks, in police commissions and others places for the purpose of preventing forgedity and crime action occurred in related with hand writing. Due to this reason a lot of research’s are done in different language writing styles by applying different algorithms with their own accuracy or performance. This research is performed on automatic hand writer identification system through CNN and SVM by considering shape, angle and orientation of characters for the purpose of preventing crime and forgedity which is not done by combination of CNN with SVM and Gabor Feature Extraction.

The research will answer the following research questions.

RQ1: What appropriate preprocessing technique is suitable for hand writer identification?

RQ2: To what extent SVM improves CNN for better recognizing hand writer's document?

RQ3: What is the performance of the proposed model?

1.4. Objectives of the study

The following are general and specific objective/s of the study.

1.4.1 General Objective

The general objective of the research is designing an automatic hand writer recognition for Amharic language by using Convolutional Neural networks and Support vector machine.

1.4.2 Specific Objective

The specific objectives of this research are listed as the following

- To review different literatures and understand the concept of writer identification system and how it is designed;
- To collect different handwritten documents of different writer to pass-through preprocessing activities.
- To build the model based on Convolutional Neural networks and support vector machine
- To evaluate the model by performance metrics
- To compare the model with related models

1.5. Scope and Limitation of the Study

1.5.1. Scope of the Study

The scope of this research lies on identifications of Amharic hand written document writers by applying machine learning approach through feature extraction and classification technique. It also includes preprocessing the hand writer's document image to their proper classification of the writer classes. The handwritten document that will serve for learning and classification purpose in this study is text independent. And the areas which is covered by in this study are

Amhara regional police commissions, Amhara regional state courts, Commercial Bank of Ethiopia Peda Branch, and Belay Zeleke Branch, Bahir Dar Preparatory school and some selected voluntary individuals are the coverage area of this paper. And this research or study takes into consideration handwriting of individuals in Amharic script only.

1.5.2. Limitation of the Study

Because of a number of reasons; This paper has its own limitation that is stated below.

- This study does not include online individual handwritings.
- Because of lack of knowledge about other languages except Amharic language scripts, the study doesn’t include or doesn’t check for other Ethiopian languages.
- Due to time constraints, the study's experiment is conducted using only a few algorithms (CNN, Gabor and SVM). Decision trees and other machine learning algorithms are not tested.

1.6. Significance of the study

As we have been discussed in the scope of this research; The target area of this research are police commission offices, Federal and state courts, Banks, Historical organizations which have historic documents and others are the beneficiary organization from this research. Police commissions also the beneficiary’s organizations from this research in terms of detecting individuals involved in criminal justice and terrorist activities linked to handwritten papers. Commercial Banks, micro and credit institutions in related with finance also are the beneficiaries from this; in authenticating individuals while paying money.

1.7. Organization of the Thesis

This work is divided into six chapters as follows. The first chapter includes an overview of the back ground and statements of the problem for the study, as well as the objectives, scope, and limitations of the study, the study's importance, and a discussion of the technique used to perform the study.

The second chapter of this research is a review of literature and related works and describes about writing system of Amharic language. Recognizing patterns related to handwritten documents, discussing different types of offline author recognition systems, the main steps to be followed by offline author recognition systems, the two algorithms selected in this study, namely convolutional neural networks and support vector machines, and the review introduced the research related to this research was conducted on a system that recognizes language authors on a global scale.

The study's methodology is discussed in Chapter three. The suggested system architecture, algorithms are used, and the overall research strategy are all given.

Under chapter Four, the details of the experiment, the results acquired from the experiment utilizing tabular and graphical representations, discussion of the experiment's results, subjective evaluation of the study, and comparison of the study with other work are presented.

Conclusions obtained from the research, as well as suggested future research directions linked to this study, are offered at the end of chapter five of this study.

CHAPTER –TWO

LITERATURE REVIEW AND RELATED WORKS

2.1 Literature Review

2.1.1.NATURE OF AMHARIC WRITING SYSTEM

Introduction

Since Amharic handwriting is the main focus of this study, this section contains explanations regarding Amharic character sets and their natures, as well as characteristics of Amharic handwriting.

One of the native African writing systems is the Ge'ez script, which dates back to AD 300. Only 26 consonants are in the original Ge'ez script and it doesn’t contains vowel indicators(Daniel Demoze and Eneyew Adugna, 2010). But later, six non-basic letters (vowels) for each of the 26 consonants were added.

As it is stated in(ALAMIREW, 2015) and (Daniel Demoze and Eneyew Adugna, 2010) these vowels were constructed by altering the structure of existing consonant symbols, resulting in a total of 182 which is (26 x 7) letters are there in Ge‘ez script. The Ge'ez script eventually gave birth to the Amharic writing system, which is today used in Ethiopia to write Amharic and other Semitic languages such as Tigre, Harari, and Gurage. (Wegman, 2010). Now let’s see the overall issues in related with Amharic Handwriting. Because this paper concerns on Amharic handwriting. In today's world, Amharic is widely spoken language. Amharic evolved from the original Ge'ez script and now has a larger character set. Following Arabic, Amharic is the world's second most widely spoken Semitic language(Negussie, 2006).

Aside from the 182 symbols in Ge‘ez script, 7 additional constants whose shapes were derived from the already existing Ge‘ez characters were added to the Amharic writing system, bringing the total number of characters in the Amharic character set to 231 (33 x 7). These what we have newly added consonants are ሸ (acquired from ሰ), ቸ (acquired from ተ), ኘ (acquired from ነ), ዠ (acquired from ዘ), ጀ (acquired from ደ), ጨ (acquired from ጠ) and ኸ (acquired from ከ). A special character designated with six non-basic forms was included in addition to the 33 basic characters to represent the sound ‘v' in Latin-based languages. As a result, in the Amharic writing system, there are 231+7 alphabetic characters(Daniel Demoze and Eneyew Adugna, 2010);((Mesay, 2003);(Shimeles, 2005)).

Derived letters	Original (Ge’ez) letters
ሸ	ሰ
ቸ	ተ
ኘ	ነ
ኸ	ከ
ዠ	ዘ
ጀ	ደ
ጨ	ጠ

Table 1: Derived Characters in the Amharic Character Set

In general, in Amharic writing system there are 33 basic characters and a total of 231 alphabetic characters. However, the character is included as a special character to represent the sound ‘v' in Latin-based languages, and it also comes in six non-basic variants. As a result, 231+7 is the number of alphabetic characters. Each of the first ten natural numbers (1 to 10) and numbers that are multiples of ten in the range 20-90 and 100, and 1000 are represented by 20 symbols in Ethiopic numerals. Other numbers are formed by adding these 20 different digits together. For example, the number 31, is written as ፴፩, which is obtained by writing 30 (፴) and 1(፩) side by side. Ethiopian Orthodox Tewahedo Church presently uses these numerals. Individuals, on the other hand, do not usually utilize these digit characters to write numbers. The characters are laid down in a table format, with 33+1 rows and 7 columns. Basic characters, which are 33+1 in number, are found in the first column. The non-basic characters, whose shapes are derived from the fundamental characters, are represented in the remaining six columns.(Negussie, 2006) and (ALAMIREW, 2015) and (Daniel Demoze and Eneyew Adugna, 2010).

In general, The Amharic character set contains a total of 306 characters. if Ethiopian numbers, punctuation symbols, and the extended character group are counted alongside alphabetic characters(Negussie, 2006).

As it is stated in (ALAMIREW, 2015)The Amharic writing system lacks a sign for zero (0), as well as decimal points and negative symbols, making arithmetic operations extremely difficult to compute. Other extended groups of characters, such as ሏ፣ ሟ፣ ሯ፣ ሷ፣ and etc. are likewise which is used to present labialization (Table 4).

ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ
ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ
ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ
መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ
ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ
ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ
ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ
ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ
በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ
ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ
ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ
ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ
ነ	ኑ	ኒ	ና	ኔ	ን	ኖ
ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ
አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ
ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ
ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ
ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ
ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ
ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ
ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ
የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ
ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ
ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ
ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ
ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ
ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ
ጰ	ጱ	ጲ	ጳ	ጴ	ጵ	ጶ
ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ
ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ
ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ
ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ
ቨ	ቩ	ቪ	ቫ	ቬ	ቭ	ቮ

Table 2: Amharic core character set (fields).

፩	፪	፫	፬	፭	፮	፯	፰	፱	፲
1	2	3	4	5	6	7	8	9	10
፳	፴	፵	፶	፷	፸	፹	፺	፻	፼
20	30	40	50	60	70	80	90	100	1000

Table 3:Amharic numerals

ሏ	ሗ	ሟ	ሧ	ሯ	ሷ	ሿ	ቈ	ቍ	ቋ
ቌ	ቍ	ቧ	ቯ	ቷ	ቿ	ኈ	ኍ	ኋ	ኌ
ኍ	ኗ	ኟ	ኧ	ኰ	ኵ	ኳ	ኴ	ዀ	ዟ
ዧ	ዷ	ጇ	ጐ	ጕ	ጓ	ጔ	ጕ	ጧ	ጯ
ጷ	ጿ	ፏ	ፗ						

Table 4፡ Amharic labialized characters Nature and analysis of Amharic characters.

As we have been discussed in the above and it is described in the work of (Yaregal, 2002)Amharic character set is grouped in two perspectives as basic and non-basic ones. 33+1 basic characters are there which is listed in the first column (Table 2) and they are called first order characters. Each basic character serves as a foundation for the non-basic characters. The characters listed in all columns of Table 3.2 above, except the first, are non-basic characters generated from their basic families. Non-basic forms are produced from basic forms by adding minor appendages (diacritic marks) to the right, left, top, or bottom in a more or less regular pattern, according to (Mesay, 2003) Some are created by adding strokes to each basic character, while others are created by adding loops or other forms of distinctiveness. Based on whatever column they belong in Table 4, these non-basic characters are referred to as second order characters, third-order characters, and so on.

i. First Order Characters

As described in the introduction part of this chapter, the first order characters of the Amharic writing system (almost distinct in shape except some of them) are the 26 original characters of the Ge'ez script, 7 derived characters, and one special character. These basic characters can be divided into five categories based on their shape similarity as it is discussed in (Yaregal, 2002); (ALAMIREW, 2015) and (Negussie, 2006). i.e.,

- characters that might have one straight 'leg' (for example የ, ተ, ቸ, ገ, ቀ, ነ, ኘ, ፐ), or
- characters that might have two straight 'legs' (such as ለ, በ, አ, ከ, ሰ, ሸ, ዠ, ዘ, ቨ),
- characters that might have three straight 'legs' (for instance ጠ, ጨ, ሐ), or
- characters that might have rounded bottom (like ሀ, መ, ሠ, ዐ, ወ), or
- characters that might have horizontal bottom line (for example ረ and ፈ).

In addition to the work of (Yaregal, 2002); (ALAMIREW, 2015);(Mesay, 2003) and (Negussie, 2006) it is stated that many basic characters are relating and have similar shapes and structures like ረ and ፈ, ጠ and ጨ, ነ and ከ, ነ and ኘ, ደ and ጀ, ተ and ቸ, ሰ and ሸ, ከ and ኸ. we prefer grouping the 26 original characters in one group and the remaining derived characters and the special character ቨ in another category as indicated below.

ሀ	ለ	ሐ	መ	ሠ	ረ	ሰ
ቀ	በ	ተ	ኀ	ነ	አ	ከ
ወ	ዐ	ዘ	የ	ደ	ገ	ጠ
ጰ	ጸ	ፀ	ፈ	ፐ		

Table 5: Group1 (Basic character)

ሸ	ቸ	ኘ	ኸ
ዠ	ጀ	ጨ	ሸ

Table 6: Group2 (Derived character)

ii. Second Order Characters

As we have seen in the above (Table 3.2) second column; A horizontal stroke is usually attached to the middle of the right side of the basic characters to give these characters and their shape.

For example, ቡ (gained from በ), ጩ (gained from ጨ), ኩ (gained from ከ), ኡ (gained from አ), ጉ (gained from ገ), etc. Some second order characters such as ሩ (derived from ረ), ዉ (derived from ወ) and ፉ (derived from ፈ) are exceptional to this rule(Shimeles, 2005)(ALAMIREW, 2015) and (Yaregal, 2002).

iii. Third Order Characters

As per we have to observe in the above table column three (Table 3.2) and discussed in the work of (Yaregal, 2002), the shape of the third order characters is mostly determined by two rules.

Rule1: Extend the right leg horizontally to the right if the basic characters have one or more legs. Examples are ሊ (from ለ), ኪ (from ከ), ዚ (from ዘ), ቂ (from ቀ) and so on.

Rule2: If the fundamental characters don't have legs, add leg and extend it like in Rule 1. Some of Letters which is follow this rule are: ቲ (from ተ), ቂ (from ቀ), ዲ (from ደ), ሺ (from ሸ), ኪ (from ከ) and so on

Only three characters in the third order do not adhere to one of these rules. These are ሪ (from ረ), ዪ (from የ), ፊ (from ፈ)

iv. Fourth Order Characters

As we have seen in the above table fourth column (Table 3.2) the fourth order characters are obtained by applying the following principles(Shimeles, 2005).

Rule1: If the base character has two or more legs, shorten the left or all legs except the one on the right. For example: ላ (from ለ), ጫ (from ጨ), ዳ (from ደ), ሻ (from ሸ), ካ (from ከ) and so on are constructed using this rule

Rule2: Extend one leg to the left for base characters (or connect a horizontal stroke running to the left to the leg). Some of the characters are: ታ (from ተ), ቃ (from ቀ), ቻ (from ቸ) and so on are constructed using this rule

Rule3: Reduce the size of the character by aligning to the upper corner and attaching a long vertical stroke to the right side of the base character if the basic character has no leg at all. Example ሃ (from ሀ), ማ (from መ), ወ (from ቀ)

However, there are also some characters which doesn’t follow the above-mentioned rules and are considered as exceptional in their group.

v. Fifth Order Characters

The fifth-order characters mostly add loops at the base of the characters' right legs. They are constructed using the following rules in particular.

Rule1: Attach a loop to the right leg of base characters with two or more legs who don't have a loop. For Example, ሌ (from ለ), ቴ (from ተ), ጤ (from ጠ)

Rule2: If the basic characters don't have legs, shrink the character by aligning it to the upper corner and attaching a vertical stroke to the right of the base characters with a loop at the bottom end. ሄ (from ሀ), ሜ (from መ), ዌ (from ወ) are constructed this way.

In Addition, the characters ዬ (from የ), ጬ (from ጨ) are derived by adding a horizontal stroke with a loop at its right end and ሬ (from ረ), ፌ (from ፈ) are formed by attaching a loop to the horizontally lying end of the base characters.

vi. Sixth Order Characters

The sixth and seventh order characters are difficult to construct rules for since they are exceedingly irregular. As a result, we categorized the characters by grouping characters with similar shapes together.

Group No	Characters
Group 1	ህ፣ ክ፣ ኽ፣ እ
Group 2	ል
Group 3	ሕ፣ ቅ፣ ት፣ ች፣ ጥ፣ ጭ፣ ፕ
Group 4	ብ
Group 5	ም፣ ሥ
Group 6	ር፣ ፍ
Group 7	ስ፣ ሽ፣ ዕ፣ ፅ
Group 8	ኅ፣ ን፣ ኝ
Group 9	ው
Group 10	ዝ፣ ዥ

Group 11	ይ
Group 12	ድ፣ ጅ፣ ጽ፣
Group 13	ግ

Table 7: Grouping the Sixth order character

vii. Seventh Order Characters

The majority of seventh-order characters are created by shortening the right one or two legs of two or more foundation characters. Some of examples are: ሶ፣ ቦ፣ ኾ፣ ዞ etc.

characters with no leg, a straight (like in the case of ሞ) or left-declined vertical line (as the case of ሦ፣ ዎ፣ ዖ፣ ፆ፣ ፖ) is connected to the bottom of the characters. Other seventh order characters such as ሆ፣ ቶ፣ ቆ፣ ኮ፣ are formed by putting a loop at the top of the base characters. Further, it is observed that the letters ሎ፣ ዮ፣ ጎ are unique in derivation of their shape than the others. In general, it can be said that the seventh order characters follow different patterns and recognizers must consider these different patterns differently.

2.2 Overview of Hand writer recognition

Topics covered in this section includes pattern recognition in relation to handwriting, different types of writer identification systems, and the major procedures involved in offline text independent writer identification systems.

2.2.1. Pattern Recognition

Pattern Recognition is a mature but exciting and fast developing field, which underpins developments in cognate fields such as computer vision, image processing, text and document analysis and neural networks. Pattern recognition is a machine learning discipline whose purpose is to classify data items based on patterns extracted from the object itself using a priori knowledge or statistical techniques. Pattern refers to the qualities or attributes of a data object that are relevant to the pattern recognition task. Identifying and sensing distinguishing patterns of objects, extracting these patterns computed from sensor observations, and classifying and analyzing the extracted information based on probabilities generalized from statistical regularities or by comparing the structure with patterns that have already been classified or described are all part of the pattern recognition system(Theodoridis et al., 2003).

As it is stated in the book of (Alder, 2001)Pattern recognition has applications in a variety of fields, including machine printed and/or handwritten character recognition, digital image processing and analysis, speech signal analysis and processing, human identification using individual faces, fingerprints, handwritten texts, or iris patterns, market prediction, disease diagnostics, and so on. Thus, pattern recognition is a research area that studies the rules and operations, as well as the design of systems that generalize knowledge from samples and make predictions. It entails a number of steps, including preprocessing activities, extracting features, and classifications, all of which require pre and post actions.

In general, it is possible to summarize in this way; i.e., Pattern Recognition is the process of distinguishing and segmenting data according to set criteria or by common elements, which is performed by special algorithms(Shimeles, 2005).

2.2.2.Offline Handwriting Writer Identification System

Individuals' handwriting must be scanned to the computer for the system to access, evaluate, and establish a likely candidate author. This technique can be used to verify the validity of handwritten emails as well as historical document analysis fields. Individual signature authentication, which is widely employed in banks for person authentication and document authentication in general, is a specific example of writer identification(M. Bulacu, 2007). It's also useful for banking and credit institutions, criminal justice, security and access control systems, and other organizations that deal with handwritten documents. As per we have discussed in the above there are different category of writer identification system. That is online and offline writer identification system. But this paper doesn't consider online writer identification system. The only concern of this paper is offline writer identification system. Now let's see the detail description of offline writer identification systems. Text dependent vs text independent systems and Constrained vs unconstrained systems.

1. Text-dependent Systems and Text-independent Systems

In addition to the above online and offline classification scheme for writer identification systems, there is another type depending on the nature of the handwriting employed for identifying purposes termed text dependent and text independent writer identification systems.

Text-dependent identification methods compare individual characters or words of known text content to a database, and so are similar to signature verification approaches but more generic in approach. As a result, text-dependent systems require the prior segmentation of relevant information by the user or by an automatic segmentation algorithm. Their advantage is that they can use their knowledge of the data's content to differentiate style from content, improving the system's accuracy. However, their fundamental disadvantage is that they are ineffective in situations where the text is unavailable, such as when comparing text documents with differing content in criminal justice systems(M. Bulacu, 2007), (Daniel Demoze and Eneyew Adugna, 2010)(ALAMIREW, 2015).

Text independent writer identification systems, on the other hand, make no assumptions about the underlying data of the handwriting, and hence are unlikely to work with text-dependent methods, which are more constrained and require similar verification text. These algorithms may identify the author of a questioned sample of handwritten document without requiring a comparison of the same characters and are unaffected by the handwritten document's text content.

In text independent identification approaches, a small amount of handwritten data is required to assure the stability and consistency of data-mined attributes. They model style information independent of content and can identify the writer based on any given text, which is commonly done by computing statistical features from huge amounts of data in order to remove abnormalities caused by specific text (M & Mary Idicula, 2011) and (Abushariah et al., 2012).

To recap, different writers compose the same material, and the diversity among writers is explicitly reflected in text-dependent writer identification systems. They merely have to match the same characters, and the writer must produce the same text as a result. Text independent identification systems, on the other hand, unlike text dependent identification systems, do not require a fixed specific handwritten text to identify the writer of a questioned handwritten sample; instead, they use and work on features obtained from unrelated to the content of the handwritten sample to identify the writer.

2. Constrained vs Unconstrained Systems

As it is stated in the work of (Shimeles, 2005) and (Fink & Ploetz, 2006) Handwriting that is constrained adheres to pre-defined writing constraints, such as all characters in the handwriting being distinct and non-touching. There are various limitations on writing styles for limited systems. Some systems, for example, may require writers to write in a distinct fashion, while others may need writers to write in a specific order of strokes. In contrast to restricted handwriting, unconstrained handwriting allows writers to write in their own handwriting, with no limits on writing style. For a system that works with unconstrained handwriting, several factors that affect text appearance will be taken into account, including writing styles, text size, writing implement utilized, writing surface, digitization, and threshold. Unconstrained systems, on the other hand, have a poorer accuracy than constrained systems, despite providing writers with more freedom.

In general, as it is adapted from (ALAMIREW, 2015) the different types of writer identification systems are represented in the diagram below.

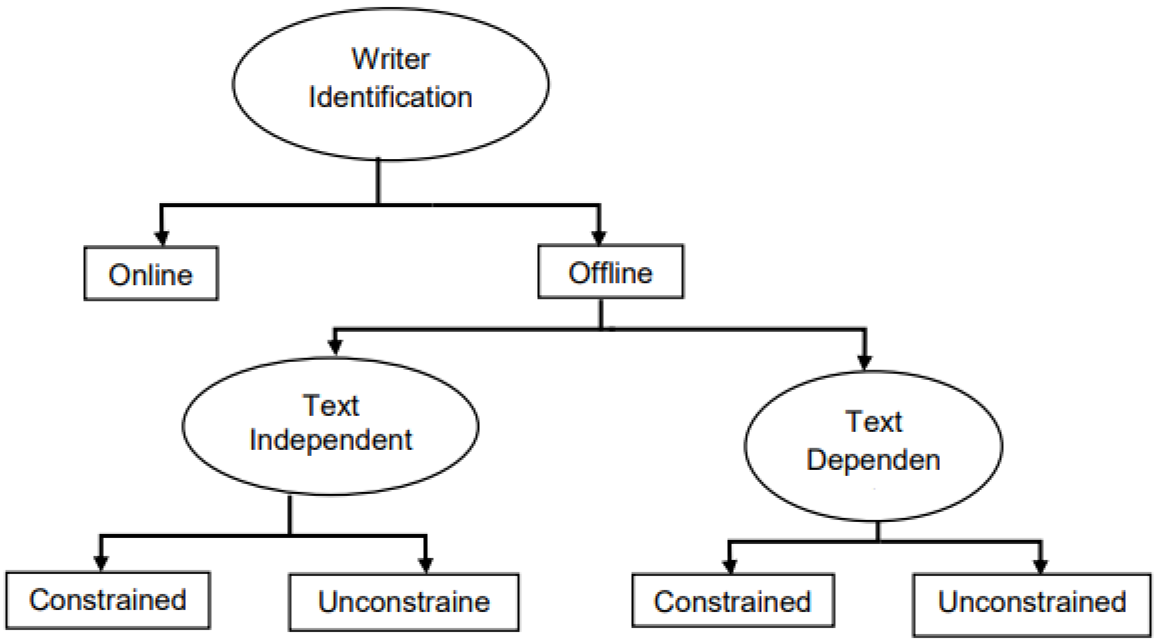

Figure 1: Categories of writer identification, Adapted from (ALAMIREW, 2015)

Major steps in offline text independent writer identification systems

Handwriting, like other biometrics like fingerprints, iris, speech, signature, face, and others, is unique to each person. Attempting to identify the author of a handwriting paper is clearly not a one-step process. Writer identification entails a series of processes that a handwriting document must go through before the author may be determined. The stages are mostly used to raise the differences between handwritten writers' signatures in order to make the identification process easier when determining the author of a questioned document.

Preprocessing, feature extraction, and classification are the three key processes in a writer identification system for handwriting documents specially for this paper, as explained below.

1. Preprocessing

The preprocessing phase improves the image over its original state, allowing for improved feature extraction. The preprocessing method to be utilized will vary depending on the sort of application for which the image is being used (ABEBE, 2011). Because noise in handwritten data can cause inconsistencies that lead to incorrect classification, the purpose of preprocessing is to reduce or eliminate these variances in order to improve the dissimilarity between/among writers, making author identification easier. They are document analysis techniques performed on scanned handwritten documents prior to identifying the writer.

Some of the preprocessing activities that have been implemented in offline writer identification systems Specially in this paper are discussed below

i. Image Acquisition

The goal of Image Acquisition is to turn an optical image (Real World Data) into a numerical data array that can be later modified on a computer. Before any video or image processing can begin, an image must be acquired by camera and translated into a controllable object. Scanners that are suited for image acquisition are used (Mishra et al., 2017).

Scanners can create picture representations in many digital formats from offline handwritten documents (paper format) in this manner.

ii. Resize Image

Resizing is altering the size of our image without cutting anything out. Resize an image is essential for the purpose of changing or minimizing the file size. Because sometimes, size does matter specially to minimizing the computation time.

Resizing can help your image fit into a certain space on a screen, such as in a blog post or social media post. Smaller images weigh less than larger ones, which can be important for making an uploading process faster or taking up less storage room on your devices. You can use the Resize tool to decrease the size and make it weigh less for the purpose of Storage and computation time(Lin et al., 2014).

iii. Noise Removal

In pattern recognition, noise refers to anything in the data that prevents the pattern recognition system from completing its work. The type of noise and the method for removing it may differ from one pattern recognition system to the next(ABEBE, 2011). The quality of the scanner or the age of the document might cause noise in the digital image of scanned papers. Furthermore, noise is caused by inaccuracies in the picture acquisition process, resulting in pixel values that do not accurately reflect the true intensities of the real scene. As a result, filtering noises is necessary before further processing the image.

iv. Segmentation

Threshold segmentation is a very popular segmentation technique, used for separating an object from its background. The scanned image was grayscale at the time of scanning. Threshold segmentation is the process of converting a grayscale or colored image to a binary image. The Threshold segmentation method aids in the separation of background and text, as well as the calculation of mathematical equations. The grayscale image is converted to pure black and white using this method (Kırlı & Gülmezo, 2012), (Ali & Suresha, 2020).

2. Feature Extraction Phase

After segmentation, feature extraction is the second most critical method of digital image processing. This phase is one of the most demanding and challenging tasks of pattern recognition systems, as it has a significant impact on the design and performance of the classifier.

It is clear that in pattern recognition, we strive to attain the correct classification rate for the relevant attributes at all times. We examine data, derive meaningful information from what we have, and rule out the impact of related or recurrent factors. Furthermore, an action to limit the dimension of information features (also known as point of interests) as little as feasible without influencing issue solving. This effort is known as feature extraction, and it creates favorable conditions for the reliability, a critical link that affects the final outcome of pattern recognition systems(Ding et al., 2012) and (Rehman et al., 2019).

Furthermore, the primary goal of feature extraction and selection is to identify a set of the most effective features for classification, i.e., compressing from a high-dimensional feature space to a low-dimensional feature space in order to create an effective classifier. In pattern recognition, all features are not used, only the good ones. Good features are constant across several patterns of the same scene, insensitive to noise, and invariant to specific transformations.

Feature extraction seeks to find patterns that are successful in class discrimination using the smallest number of characteristics possible. In offline writer identification, several features of the texts may be essential depending on the requirements and the approach of the process used.

According to the researchers, features in writer identification do not correlate to a single value, but rather a probability distribution function (PDF) generated from handwriting images to represent writer individuality. Texture traits and allograph (character shape) attributes are commonly documented in the literature for text independent offline writer identification.

The goal of image feature extraction is to improve object recognition by extracting new characteristics from raw pixel data.

3. Classification

After learning the distinguishing characteristics, classification is carried out. Feature learning is made up of multiple layers that are placed on top of each other. This section contains a description of each operation used for learning features and classifying into preset classes at each phase. It is described in chapter 3 in detail.

2.3 Evaluation Technique

The general steps for creating a machine learning classification model include data collection, data exploration, pre-processing, feature extraction, and classification approaches. However, before we conclude the classifier model, we must confirm that it works properly, which is known as model evaluation.

The performance of the proposed solution or model has been evaluated using a variety of performance indicators. Accuracy, precision, recall, and the f1-score are among the metrics frequently used to assess the effectiveness of proposed solutions.

Accuracy is defined as the percentage of true positives (including true positives and true negatives) in comparison to the entire population. If the class is not balanced, accuracy may mislead the model's quality.

$$\text{Accuracy} = (TP + TN) / (P + N) \tag{2.1}$$

Precision is defined as the ratio of true positives to total positives. It can be stated mathematically as:

$$\text{Precision} = TP / P \tag{2.2}$$

Recall or sensitivity is the fraction of true positives compared to the total amount of true or right data. It assesses the model's ability to avoid false negatives(Adam & Josh, 2017). It's also known as the hit rate or true positive rate.

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (2.3)$$

F1-score is the weighted average of precision and recall. Precision and recall both make an equal proportion to the F1-score.

$$\text{F1-score} = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall}) \quad (2.4)$$

For all of the abovementioned performance measures, micro-average, macro-average, and weighted-average can be derived and used for further research.

Macro-average precision or recall is just the average of the precision and recall (respectively) of the model on different classes.

$$\text{Macro-average precision} = (\text{P1} + \text{P2} + \ldots + \text{PN}) / \text{N} \quad (2.5)$$

$$\text{Macro-average recall} = (\text{R1} + \text{R2} + \ldots + \text{RN}) / \text{N} \quad (2.6)$$

The individual true positives, false positives, and false negatives for each class are added together to determine the micro-average precision or recall.

$$\text{Micro} - \text{average precision} = \frac{\text{TP1} + \text{TP2} + \ldots + \text{TPN}}{(\text{TP1} + \text{TP2} + \ldots + \text{TPN}) + (\text{FP1} + \text{FP2} + \ldots + \text{FPN})} \quad 2.7$$

$$\text{Micro} - \text{average recall} = \frac{\text{TP1} + \text{TP2} + \cdots + \text{TPN}}{\text{TP1} + \text{TP2} + \cdots + \text{TPN}) + (\text{TN1} + \text{TN2} + \ldots + \text{TNN}} \quad 2.8$$

2.4 Related works

This section seeks to give research on Arabic, Chinese, Latin and Ethiopic scripts' writer identification methods. These three scripts were chosen because they have significant differences in nature that will reveal which technique is best for which writing system.

Different writing systems have different handwriting styles. The difference could be in the quantity of letters, their shape, the degree of similarity between characters in the same set that make up the writing system, and a variety of other characteristics. When developing a handwritten identification System, the method and approaches used should take into account the nature of the writing system. The use of classification algorithms varies mostly due to the type of the hand writer’s text in the system.

2.4.1 Arabic Handwriting Writer Identification System

The Arabic script is an abjad writing system that is employed in a variety of Asian and African languages. The Arabic alphabet contains 28 letters that are used to write the Arabic language. The Arabic writing method is written in a cursive style from right to left(Hellmuth, 2013).

As it has been stated on (Marius Bulacu et al., 2007)On offline Arabic handwriting, text-independent writer identification and verification; The objectives of the study was to see how well texture and allograph traits on Arabic handwriting performed in terms of text-independent writer identification. As it is described in the study, the attempt to discover the writer of a handwritten document in identification systems is vulnerable by both between-writer and within-writer variable issues. As a result, elements in automatic writer identification are critical to maximizing differentiation between different writers (between writers) while maintaining constant over handwritten by the same writer (within writer).

The research focused on two different layers of feature extraction and analysis: texture and allograph characteristics. The handwritten document is treated as a texture for texture level characteristics, and the features studied are directional & hinge contour, directional cooccurrence, and run length probability distribution functions. The writer is considered a shape generator for allograph level features, and the feature studied is ink-blob shapes (graphemes), which are unique to each writer and are computed using a common shape codebook derived by clustering(Marius Bulacu et al., 2007).

The writer identification system was done using nearest neighbor classification in a leave-one-out strategy, in which one sample from the handwritten samples used for identification is chosen as the query, and the remaining handwritten samples are ordered with increasing distance from the query sample using a selected feature. The writer of the first ranked (Top-1 hit size in this example) handwritten sample is presumed to be the writer of the inquiry handwritten document based on this sorted list. For the purposes of the study, larger hit sizes may be required. The researchers employed the chi-square (x^2) distance to match attributes of a questioned sample with samples of known authorship.

According to the results of the experiment, the contour-hinge probability distribution function works best, with top-1 outcomes of 82 percent and top-10 results of 97 percent. The grapheme probability distribution function is the next well-performing feature, with a result of 60% for top-1 and 90% for top-10. The same performance was obtained for the three clustering algorithms (k-means, k-som1D, and ksom-2D) utilized for constructing the grapheme codebook for the study. Furthermore, when a combination of directional, grapheme, and run length features is integrated and fused, the best result is 88 percent for top-1 and 99 percent for top-10.

To summarize, improving the performance rate for writer identification can be accomplished by mixing features from several feature groups, such as textural and allograph-based features.

2.4.2 Chines Handwriting Writer Identification System

Two studies are done in related with chines writer identification system are along with short explanations.

Stable Spectral property of texture photos is used to identify Chinese handwriting as it is studied on the work of (Yan et al., 2009).

Chinese and some other Asian languages are written using the Chinese character. The characters number in the tens of thousands, despite the fact that the majority are minor graphic variations. There are almost 50,000 characters in the Chinese language. Pictographs, ideographs, and phonetic compounds make up the Chinese character, which is one of the most complicated. Chinese is a logographic writing system, which means that each sign represents a word or a smallest unit of meaning(Leng & Shamsuddin, 2010).

The study's goal was to use a spectral feature extraction method based on quick Fourier transformation to improve writer identification for huge samples of Chinese handwriting. As a result, texture analysis has become the most extensively used method for identifying handwriting in text-independent writer identification systems. Textural approaches for identifying writers have been tested and shown to be more effective for small handwritten samples.

The research proposes a method for identifying the author of Chinese handwriting based on stable spectral characteristics texture, which includes scanning, normalization, feature extraction, and classification activities. For the study, 200 handwriting samples were obtained from 100 people, each of whom was allowed to write two different handwritings, one of which was utilized to create the sample database and the other for testing. Regardless of the material or writing technique employed, these handwritings must have at least 200 Chinese characters. When it comes to top-10 handwritings, one texture image achieves 87 percent accuracy.

It is also stated that randomness can affect the features of a single texture image, such as writing randomness, tools and environments used for writing, the state of the writer at different times;

randomness in the content of handwriting, resulting in different textural features; and randomness in the position of characters that make up the texture image.
The Fast Fourier Transform feature extraction technique, on the other hand, has reduced randomness effects. The experimentation was further double-checked by increasing the number of writers and texture images used in the study. Using 20 texture images, Euclidean distance classifiers got a maximum result of 98 percent accuracy for handwritten samples of 100 writers, and a maximum result of 95 percent accuracy for handwritings collected from 500 individuals.

To summarize, employing multiple textural images from handwritten samples instead of single texture images has improved the writer identification performance to a great extent, given that handwritten samples of persons contain appropriate amounts of written Chinese characters.

Identification of Chinese Handwritten Writers Using Structure Features and an Extreme Learning Machine was the second study(Tan et al., 2013).
By proposing the Chinese Structure Feature (CSF) technique for feature extraction and combining it with an extreme learning machine, a new scheme for writer identification of Chinese handwritten documents is offered (ELM). Because there are repetitive phases to obtain the ideal values of input and output weights, the back propagation process is slow. In their research, they used an efficient and practical learning mechanism dubbed extreme learning machine for single hidden-layer feed-forward neural networks. Instead of utilizing iterative approaches like back propagation, ELM learns the input and output weights by directly calculating the generalized inverse matrix of the neural network's hidden layer output matrix. When compared to iterative algorithms, it achieves the best generalization performance, runs faster, and has the least weight norm(Tan et al., 2013).

Each character's features are taken directly, and the stroke shapes and structures are used for handwriting identification. Two structures, the bounding rectangle and the TBLR quadrilateral, are used to extract features. Nine Chinese Structure features, seven TBLR quadrilateral structure characteristics, and four more Chinese structure features are recovered from the first structure. The study extracts a total of twenty features for Chinese handwritten identification. The paper goes over these qualities in detail(Tan et al., 2013).
Two Chinese handwriting databases were used in the experiment (Tan et al., 2010) and (Kim & Kim, 2003) were used.

In (Tan et al., 2010) database A total of 950 Chinese characters were created by 245 writers. That of KAIST Hanja1database contains 783 commonly used Chinese characters produced by 200 different authors. Their research compares the overall time spent learning ELM to earlier studies that used back propagation (BP) and SVM. The average total time of ELM, SVM, and BP as it is stated in the work of (Marti et al., 2001) are 1s, 31s, 34s seconds respectively. It also compares and presents CSF traits with those found in previous study done using Gabor (Said et al., 2002), (Chen et al., 2008), (Xu et al., 2011) Features. It is found that employing top-20 databases, their approach achieves 98.5 percent and 95.4 percent for SYSU and Hanja1 databases, respectively. According to the study, this outcome is better than the performance obtained utilizing other feature extraction strategies.

To summarize the review, the combined effect of Chinese structural feature and extreme learning machine resulted in a better identification performance than the existing approaches employed for the problem of Chinese writer identification. This combination approach can also be used to identify writers in circumstances of multilingual handwriting, such as western handwriting.

2.4.3 Latin handwriting Writer Identification Systems

Three writer identification techniques based on Dutch handwritten papers and one study based on English handwritten documents are discussed and described in this section.

As it is stated on (Van Der Maaten & Postma, 2005) studied on enhancing automated writer recognition. The objective of the research was to overcome the constraints of statistical and model-based approaches to improve automatic writer identification. When forensic professionals assign handwritten writing to persons, such as criminal suspects, the results are unreliable. Artificial intelligence provides us with a variety of options for automating the assignment of handwritten text to different writer classes.

Based on the study of (Van Der Maaten & Postma, 2005) The authors provide two basic methodologies for identifying and analyzing writers in handwritten papers. The statistical approach and the model-based approach are the two options. As previously indicated, the statistical based strategy for the first approach entails a statistical analysis of features collected from handwritten samples. The model-based technique, on the other hand, involves the use of pre-defined models of minuscule handwriting strokes (called graphemes). But they all have one thing in common: feature extraction and classifying stages.

Run length distributions, slant distributions, entropy, and edge-hinge distributions are examples of so-called statistically based characteristics, and one can see an overview of them found at(Marius Bulacu et al., 2003). Edge hinge distribution is a characteristic that describes how a writing stroke changes direction in handwritten text. They discovered that the edge-hinge feature's fundamental shortcoming is that it only examines changes in direction on a single scale, rather than several scales. The researchers propose two new multi-scale features to overcome the problem of edge hinge feature. The first is a variation on the edge-hinge distribution feature called edge-hinge combination, which combines edge-hinge distributions formed with different fragment lengths (i.e., window sizes). The second is employing wavelet characteristics(Van Der Maaten & Postma, 2005).

A codebook of handwriting stroke models is used in the model-based method to feature extraction (graphemes). For a handwritten sample, graphemes are retrieved first, and then graphemes from the codebook contour are matched using Euclidean distance. These handwritten text strokes are effective qualities for solving the writer identification problem.

However, creating a grapheme codebook takes a lengthy time, which is discovered to be the fundamental drawback of the model-based approach. Rather than training, this challenge is solved by constructing a codebook via random selection(Van Der Maaten & Postma, 2005).

As a result of the statistically based approach employed for writer identification, the edge hinge feature performs best, and this performance can still be increased further by using edge hinge combination features with varying window widths. The time it takes to produce a grapheme code book using Kohonen SOFM can be reduced by employing random selection to create an ideal model in the model-based method. Furthermore, combining these two algorithms outperforms independent attempts at identifying the author of a handwritten work.

The second study is(Marius Bulacu et al., 2003) studied on Edge-based directional cues are used to identify the writer. The study's goal was to assess the efficiency of edge-based directional probability distributions features and compare them to non-angular features.

Five statistically based feature extraction methods for writer identification are described in this paper. Edge-direction distribution, edge-hinge distribution, and run-length distribution are the three; Entropy and Autocorrelation are the fourth and fifth respectively.

This work (Marius Bulacu et al., 2003)employed the Firemaker dataset for their research, just like (Van Der Maaten & Postma, 2005).The study only used a portion of the dataset, specifically the first and second pages. Because there were 250 writers, each of whom had two pages of handwritten content, a total of 500 pages of handwritten text was examined for testing purposes. The features listed above were taken from these handwritten pages and turned into a feature vector.

The chosen features were evaluated using nearest neighbor classification with Euclidean distance in a leave-one-out method. The results of the experiment are reported for hit list sizes ranging from one to twenty. The maximum performance for edge-based features, such as edge-direction distribution and edge-hinge distribution, was 83 percent and 90 percent for rank of twenty, respectively. Furthermore, for this hit list size, a performance of 97 percent was attained for the combined effect of the features. An experiment is also carried out to see how well the features function when the amount of handwritten text is reduced by using the entire page, half page, and initial line of the handwritten page(Marius Bulacu et al., 2003).

To summarize, the edge hinge distribution feature outperforms the other features studied, and fusing features produce superior outcomes. When a big amount of handwritten text is given, the highest performing edge distribution-based features still outperform other features when a limited amount of handwritten text is supplied, according to the results of the experiment.

The third paper which is done for Latin language script hand writer identification is(Marti et al., 2001) studied on Writer identification using text line based features. The goal was to look at elements that can be retrieved from handwritten text lines in order to identify people by their handwriting. The paper assumes that the same person created all of the segmented text line images.

Twelve features are extracted from the text line image, most of which are based on observable writing characteristics such as the height of writing zones (upper, middle, and lower zones), characters width, writing slant, and text legibility, and then a feature vector is formed. The three writing zones are determined by the top line, upper baseline, lower baseline, and bottom line. And, depending on the writing zones, the first six features are derived. The breadth of the writing yields two additional features, and the qualities of the slant of the writing yield two other features. The remaining two features are discovered using fractal geometry, bringing the total number of features to twelve. For a detail description of these features, you can see the paper(Marti et al., 2001)

For writer identification, two classification approaches are used: a k-nearest neighbor classifier and a feed forward neural network. When it comes to KNN, the value of k is a parameter that can range from 1 to 25. The distance measure is the other parameter, and in this case, the distance between feature vectors of the test sample and feature vectors of the training handwritten text line images is measured using a commonly used Euclidean distance function. The feed forward neural network is the second classification method used. One hidden layer of the network includes 5 to 25 neurons. The number of input and output neurons of the network are determined by the dimension of the feature vector of the text line picture and the number of writers, respectively. A backpropagation approach with a learning rate of 0.1 is employed to train the network, and the connection weights between layers are adjusted during training.

100 different pages of text written by 20 different writers were employed in the tests, which were acquired from the IAM database(Marti & Bunke, 2003). And each writer was given five distinct text pages to work on. According to the study's findings, the best result in the KNN classifier is 87.8%, which is attained when only one nearest neighbor is considered for a subset of feature sets. When all available feature values are used in the instance of a neural network, the best recognition rate of 90.7 percent is obtained for 20 hidden neurons.

To sum up, the paper looked at text line-based aspects of a handwritten text that can be found anywhere between word and page level features. Many writer identification publications have used the k closest neighbor classification method, but this paper has proven that artificial neural networks might be used in addition to KNN to identify people based on their handwriting.

2.4.4 Ethiopian Handwriting Writer Identification System

As far as in my searching only two studies was done in related with Amharic Handwriting hand writer identification system in Ethiopia, those researches are (Daniel Demoze and Eneyew Adugna, 2010) and (ALAMIREW, 2015).

As it is stated in the first work(Daniel Demoze and Eneyew Adugna, 2010) The goal of this study is to use Gabor Energy and Connected Component Contour characteristics to create a handwriting identification system for Ethiopic handwriting. It proposes an off-line Ethiopic writer identification method that is not dependent on text.

Gabor Energy and Connected Component Contours are the two methods used to extract features. Handwriting is supposed to be a texture picture, and writer identification is assumed to be a texture classification in the first technique. The picture of a document was first preprocessed for this purpose, and then a set of Gabor filters with eight equidistant orientations and three spatial frequencies were applied, yielding 24 filtered images. The Gabor-energy approach is then used to extract features from these filtered images. For the study, 25 people were chosen at random and asked to copy a desired text on two A4 pages in their normal handwriting. In both procedures, the features are extracted and then the x^2 distance measure method is utilized to identify them. Experiments show that employing Gabor energy characteristics, 93 percent correct identification in a hit list of size 3 and 76 percent correct identification in a hit list of size 1 is possible. Using connected component contours, 96 percent of hit list 3 and 92 percent of hit list 1 are correctly identified.

To summarize, the study is the first attempt to solve the problem of identifying people based on their Ethiopic handwriting. As many of the publications examined here show, acceptable results may be obtained utilizing the feature extraction approaches used, and greater performance can be attained by combining them.

The second is by (ALAMIREW, 2015)studied on writer identification system for Amharic handwritten documents.

The goal of this study is to create a system that can identify people based on their handwritten Amharic documents. To accomplish so, 250 handwritten texts from 50 people were collected, and after preprocessing, four key properties were extracted: edge hinge distribution, graphemes, autocorrelation, and entropy. The study's classification algorithms were k-nearest neighbor and artificial neural network, which were utilized in separately and in combination.

This paper uses MATLAB to build the system. On the edge hinge distribution feature, an experimental result of 95.6 percent was reached for a hit list size of 10 utilizing Manhattan distance measure on nearest neighbor algorithm. On the artificial neural network, a one hidden layer network with 30 neurons has an accurate identification performance of 30%. Combining these two methods for the Amharic writer identification system yielded 96.4% accuracy over a nearest neighbor utilizing a Manhattan distance hit list size of ten and a neural network with two hidden layers each containing 40 neurons. Additionally, the subjective rating yielded an average result of 82%.

S. No	Language	Author's	Objective	Feature Extraction Method	Classification	No of Writers	Sample Size	Achievable performance
1.	Chinese	Yan et al., 2009)	To improve writer identification of large handwritten samples using spectral features based on Fourier transformation.	Spectral features using fast Fourier transform	Euclidean distance	100	200	- 87%, top-10, 1 texture
						100		- 98%, top-10, 20 textures
						500		- 95%, top-10, 20 textures
2.		Tan et al., 2013	To propose approach for Chinese handwritten using Chinese structure features & extreme learning machine.	Structure features	Extreme learning machine	200	783	- 95.4%, top-20
						245	950	- 98.5%, top-20
3.	Arabic	Marius Bulacu et al., 2007	To evaluate performance of texture and allograph features on Arabic writer identification system.	Textural and allograph features	Nearest Neighbor, chi-square	411	2055	- 99%, top-10
4.		(Somaya Al-Maadeed, 2012)	To automate writer identification process using handwritten images and offer a computerized analysis of individual handwritings.	Edge based, invariants, word measure	Nearest Neighbor	100	32,000	- 98%, top-10

5.		(Gazzah & Amara, 2008)	To present a method for writer identification of offline Arabic handwritten based on combined effect of global structural effects.	Structural and textural features	ANN and SVM	60	180	- 93.76%, SVM - 94.7%, ANN
6.	Latin	(Van Der Maaten & Postma, 2005)	To improve automatic writer identification by overcoming the limitations of statistical and model-based approaches.	Statistical and model-based features	Nearest neighbor, chi-square distance	250	500	- 81%, top-1
7.		(Marius Bulacu et al., 2003)	To evaluate effectiveness & compare performance of edge-based directional features with non-angular features.	Edge based features	Nearest neighbor, Euclidean distance	250	500	- 97%, top-20
8.		(Marti et al., 2001)	To present a system for writer identification from lines of text and study features for identifying individuals based on their handwriting.	Text line-based features	KNN with Euclidean distance and ANN	20	100	- 90.7%
9.	Amharic	(Daniel Demoze and	To develop hand writing writer identification system	Gabor & connect ed component	Nearest neighbor, Chi-	25	50	- 93%, top-3, Gabor energy
								- 96%, top-3,

		Eneyew Adugna , 2010	for Ethiopic handwriting samples.		square distance			connected
10.		Lamesginew Andargie 2015	To develop a text independent and offline writer identification system	Textural Feature Extraction and Allograph Feature Extraction	KNN and ANN	50	250	- 81.2%, hit list size of 1, Euclidean, Combination of ANN & KNN - 96.4 % hit list size of 10, Manhattan, Combination of ANN & KNN

Table 8:Summary of literatures reviewed on writer identification systems

To sum up: It is well known that many languages are spoken today all over the world. Every language has its own writing style. As a result, each language adopts a different threat to the so-called writer identification problem, based on its language characteristics. As a result, the problem of identifying the author of a handwritten document varies by language. An attempt is made to review various literatures relevant to writer identification challenges for various handwritten languages such as Chinese, Arabic, Latin and Amharic. Different methodologies are employed for their study because writing styles differ from language to language. As it is reviewed, many preprocessing activities and feature extraction strategies are applied in different researchers which is conducted. However, the majority of papers use nearest neighbor with Euclidian and chi-square distance functions for classification. In addition to KNN, neural network classification methods are used in a variety of different applications. As a result, for the purpose of this study, we were address the problem of identifying the author of a particular handwritten document in the Amharic language, for this reason multiple approaches in preprocessing activities, feature extraction techniques, and classification strategies were considered(Daniel Demoze and Eneyew Adugna, 2010)(ALAMIREW, 2015).

CHAPTER-THREE

METHODOLOGY

3.1 Overview of the Proposed System

In this chapter, detailed description of the proposed system or model for identification of writers is discussed.

As we have discussed in the above; this study does not use any ready-made datasets, which means we are collecting data from relevant sources and we pass the different steps. due to this reason, the thesis includes preprocessing, feature extraction, and classification phases. That means for obtaining the appropriate or good feature from hand writers document; It passes via a series of steps starting from preprocessing such as resizing an image into appropriate image size, Noise reduction, threshold segmentation of the image (ROI), feature extraction and learning to classification into predefined classes. Classification mainly encompasses three major phases; these are training phase, testing phase and validation phase. The general working procedures i.e., Preprocessing, feature extraction, and classification are the three components of the proposed system. We normalize the image to a standard size through image resizing. The extraction of an area of interest (ROI) is done via threshold segmentation. We employ Gabor filtering and CNN for feature extraction documents written by hand. Convolutional neural networks and Support Vector Machines are used to classify the data. Classification encompasses three main phases: training, validation, and testing phase. Finally, a Softmax is used for classifying into a specific class in CNN. There is also SVM which are used for classification purpose in this model by feeding the features which is called CNN and Gabor feature extraction features.

The proposed model is shown below.

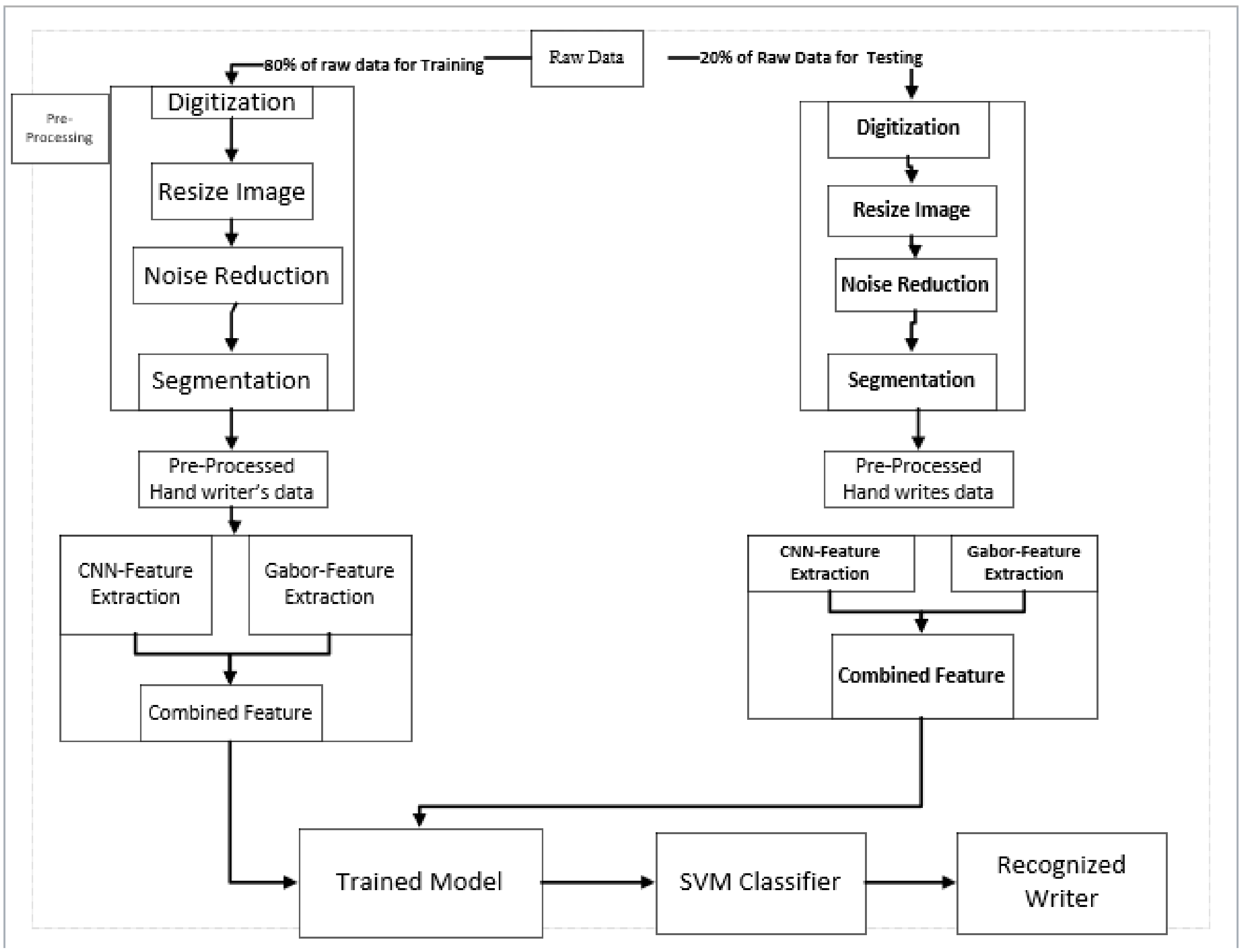

Figure 2:System architecture

The validation phase follows similar procedures with the training or testing phase. It should follow the same way, otherwise, it may produce inaccurate result or will generate error message.

3.2 Techniques for Writer Identification

3.2.1. Convolutional Neural Network

Each unit (neuron) in a fully connected layer is linked to every other unit in the previous layer. Each unit in CNN, on the other hand, is connected to a tiny number of units from the previous layer. Furthermore, all units are linked to the previous layer in the same way, with the same weights and structures. In at least one of their layers, CNNs use convolution rather than ordinary matrix multiplication. Convolution is the sum of two matrices after element-wise multiplication. The basic CNN architecture is given in figure 3 below.

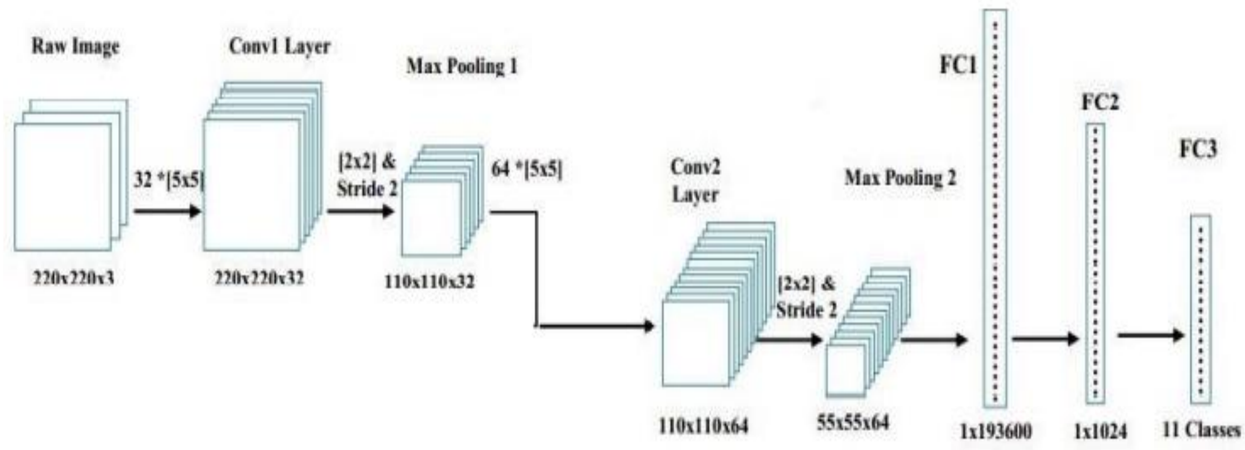

Figure 3:Basic CNN architecture

As a convolution operation, nearly all machine learning and deep learning libraries use the simplified cross-correlation function. Convolution (denoted by * operator) over a two-dimensional input image I and two-dimensional kernel K is defined as:

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n K(i + m, j + n)\, I(m, n) \qquad (3.1)$$

The kernel K (tiny matrix) sits on the top of the big image I (big matrix) and slides from left-to-right and top-to-bottom, applying a mathematical convolution operation at (x, y)-coordinate of the big image I. CNNs are able to learn kernels (filters) that can detect edges in the lower-level layers of the network and then use the edges as building blocks to detect high-level objects (features). Local invariance (through pooling layers), compositionality, and shared weights are three fundamental advantages of CNNs(Gonzalez, 2007)(Rosebrock, 2017). Local invariance used to detect an image as containing a particular object regardless of its spatial location. When designing CNNs, it should be invariant of translation, Rotation, scaling, and a variety of additional options are available. A CNN computes the same picture characteristics in all spatial areas(Farnham et al., 2017).

Compositionality is utilized to learn more detailed information from the network's deeper layers. Each filter assembles a patch of lower-level characteristics into a higher-level representation on the fly. Edges from pixels, forms from edges, and complicated features from shapes could be learned by a network. This feature is what makes CNNs so effective in computer vision.

A simple notion is to apply the same weights for all neurons in the hidden layers if we wish to move away from pixel representation in the future by gaining the capacity to recognize the same feature regardless of where it is positioned in the input image. After that, each layer will learn a set of latent properties obtained from the image that are position independent(Gonzalez, 2007).

3.2.2. Building Blocks of CNN

Many layers are used to build CNNs such as convolutional, activation, pooling, fully-connected and dropout layers. The following is the detailed description of each layer.

Convolution: takes raw data or a feature map output from another convolution as input, applies a convolutional kernel (filter), and generates a feature map. When compared to the input image, filters are modest in terms of spatial dimension (but extend throughout the whole depth of the volume) and nearly invariably square. Connecting a neuron to all other neurons in the previous layer is typically impracticable for images with huge spatial dimensions. When using CNNs, we only link each neuron to a small portion of the input volume. we call this local connectivity and the size of this local region the receptive field of the neuron.

If the receptive filed is of size 5 x 5, then each neuron in the convolution layer will connect to a 5 x 5 local region of the image for a total of 5 x 5 x 3 = 75 weights (if the image has a depth of 3 (one for each RGB channel).

As it is stated in the work of (Gonzalez, 2007) The depth, stride, and zero padding size are the three parameters that determine the size of an output volume. The depth refers to the collection of filters that all look at the same (x, y) point in the input. The depth of the convolution layer determines how many neurons (filters) are connected to a specific area of the input volume. If the input volume is 8 x 8 x 32 and the receptive field is 5 x 5, each convolution layer neuron will connect to a total of 5 x 5 local regions of the image, for a total of 5 x 5 x 32 = 800 connections to the input volume.

The stride specifies how many pixels in the x- and y-coordinates are skipped. It regulates the filter's spatial mobility across the picture or feature map. It introduces the size of skips in the filter's application. Smaller strides result in higher output volumes and overlapping receptive fields, whereas larger strides result in smaller output volumes(Farnham et al., 2017). As shown in figure 4 the size of output volume is reduced much when using stride size of 2.

12	35	117	209	9
56	123	7	38	136
235	64	111	19	23
67	203	45	182	77
3	55	101	59	221

0	0	1
0	1	0
-1	0	0

Figure 4:Left: a 5 x 5 input image. Right: a 3 x 3 kernel (filter)

5	152	-64
4	-54	110
311	9	104

5	-64
311	104

Figure 5:output of convolution with stride size of 1 (left) and 2 (right)

When applying a convolution, zero-padding means padding the original image's borders with zero to retain the original image size. The spatial dimensions of the input volume would drop too quickly if zero-padding was not used. Table 6 shows the results of applying a 3 x 3 filter to a zero-padded 5 x 5 image on the left, and the results of applying a 3 x 3 filter to the original image on the right. As can be seen, zero-padding aids in the preservation of the original 5 x 5 image's spatial dimensions.

0	0	0	0	0	0	0
0	12	35	117	209	9	0
0	56	123	7	38	136	0
0	235	64	111	19	23	0
0	67	203	45	182	77	0
0	3	55	101	59	221	0
0	0	0	0	0	0	0

Figure 6:Applying zero-padding (of size 1) to the image on figure 4 left

12	-21	-6	202	-29
91	5	152	-64	117
358	4	-54	110	-159
131	311	9	104	18
206	100	283	136	221

5	152	-64
4	-54	110
311	9	104

Figure 7:Effect of zero-padding on output image size

Activation layers are not considered technical layers because no parameters are learned in them. Due to the fact that the activation function is applied element-by-element, the output is always the same as the input dimension(Rosebrock, 2017). It is usual practice to use nonlinear activation functions after linear layers. The most extensively utilized activation function in CNNs is the Rectifying linear unit (ReLU)(yoshua bengio, 2019). Deep CNNs that use ReLU train multiple times quicker than those that use other activations like tanh. To avoid saturation, ReLUs do not require input normalization(Gonzalez, 2007).

Activations are done in-place so there is no need to create a separate output volume. ReLU outputs the maximum of zero, and the number (image pixel). i.e., max (0, x). figure 8 right below shows the output of applying ReLU to a 3 x3 input volume (figure 8 left).

5	152	-64
4	-54	110
-11	-9	104

5	152	0
4	0	110
0	0	104

Figure 8:Application of ReLU

$$ReLU=F(x)=Max\ (0, x) \tag{3.2}$$

As we have seen in the following summary table below, we used three convolutions and flattens.

CNN feature Hyperparameters		Value we used for CNN feature extraction
Input shape		224,224,1
Convolution one	filter	16
	Stride	1
	Interpolation size	32
	Kernel size	3,3
	Activation	Relu
Convolution two	filter	32
	Stride	1
	Interpolation size	32
	Kernel size	3,3
	Activation	Relu
Convolution three	filter	64
	Stride	1
	Interpolation size	32
	Kernel size	3,3
	Activation	Relu
Flatten	1-dimession feature extracted here	

Table 9: CNN Hyperparameters

Pooling layers: used to assist avoid overfitting by gradually reducing the dimensionality of the input data as a result, the number of parameters and the quantity of processing in the network are reduced. It takes a local receptive field and replaces the nonlinear activation function with the max, min, or average function at each section of the field(Bharath Ramsundar & Zadeh, 2017).

The most frequent type of pooling layer is max pooling, which is utilized to lower the spatial size of the CNN design. If our pool size is 2 x 2 and we apply max pooling, then we keep only the maximum or the largest value in each 2 x 2 block regions and the height and width of the input volume will be reduced by a factor of 2. However, we can further reduce the dimensionality of the input volume by increasing the stride size. Table 9 below depicts the effect of max pooling with a stride size of 1 (top-right) and 2 (bottom-right) while applied on a 4x4 input volume (left).

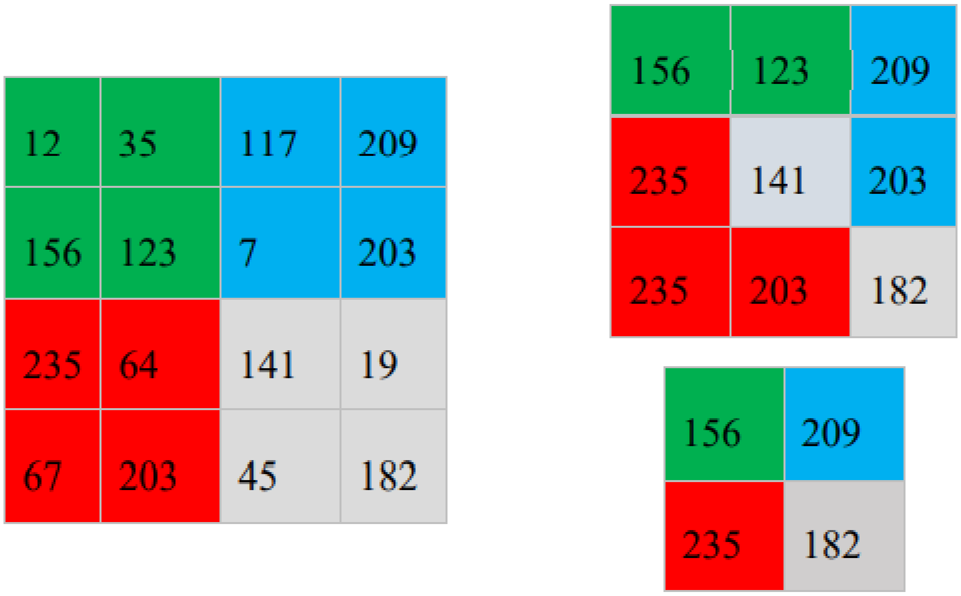

Figure 9:Effect of max pooling on different stride size

When using max pooling with a large stride size (such as 2), the input volume's spatial dimensions are drastically reduced, with 75% of the preceding layer's activations being discarded.

Fully-connected layers: This layer's neurons are entirely coupled to the prior layer's neurons. They have both normal and hyperparameters for the layer. Class scores are computed using fully connected layers, which will be utilized as the network's output(Adam & Josh, 2017). They are used at the conclusion of the network before the classifier is applied (usually Softmax classifier).

Dropout is a technique for turning off a proportion of units in a layer during training by setting their values to zero, and then turning them back on during testing by setting their values to one. It's a regularization form that forces the network to spread learnt representations over all neurons, forcing it to learn a representation that works after the dropout(Farnham et al., 2017). Dropout reduces overfitting by changing the network architecture deliberately during training. When a specific pattern is provided, it ensures that every single node is activated. Dropout eliminates the network's reliance on small units of neurons.

3.2.3. CNN Architecture

The CNN architecture is created by stacking multiple layers (convolutional, pooling, activation, or fully connected layers) in a certain way. Following are shown the successful CNN designs that have been applied in the fields of image processing and Classification.

AlexNet

AlexNet won the ILSVRC-2012 contest to classify 1.2 million images into 1000 different classes by obtaining a top-5 test error rate of 15.3 percent, compared to 26.2 percent for the runner-up(Gonzalez, 2007).

The network, which has 60 million parameters and 650,000 neurons, is made up of eight learned layers, including five convolutional layers (some of which are followed by max-pooling) and three fully connected layers, concluding in a 1000-way Softmax distribution over 1000 class labels. Every convolutional and fully connected layer's output received non-linear ReLU activation. The input dimension of the network is 150,528 (224 x 224 x 3). With 96 kernels of size 11 x 11 x 3 and a stride of 4 pixels, the first convolutional layer filters the input image.

The pooled output of the first convolutional layer is fed into the second convolutional layer, which filters it with 256 kernels of size 5 x 5 x 48. There are no pooling or normalizing layers between the third, fourth, and fifth convolutional layers. Each of the completely connected layers contains 4096 neurons.

VGG

VGG places second in the ILSVRC-2014 contest to classify 1.2 million photos into 1000 different classes, with a top-5 test error rate of 7.3 percent. In addition, VGG won first place in the localization competition. The network is made up of sixteen layers, thirteen of which have been learned.three fully connected layers with the final 1000-way SoftMax that produces a distribution over the 1000 class labels, convolutional layers with a filter size of 3 x 3, five pooling layers that follow some of the convolutional layers, and five pooling layers that follow some of the convolutional layers ReLU nonlinearity was used to apply all learned layers(Yenegeta, 2020).

Their key contribution was a detailed analysis of increasing-depth networks utilizing an architecture with extremely small (3 x 3) convolution filters convolved with the input at every pixel with a stride size of 1. It also improves on the state-of-the-art by increasing the weight layer depth to 16-19. By lowering the number of parameters and introducing additional nonlinear rectification layers, small convolutional filter sizes (such as 3 x 3) have gained a significant advantage over big convolutional filter sizes (such as 7 x 7) as employed(Yenegeta, 2020).

GoogLeNet

GoogLeNet won the ILSVRC-2014 challenge to classify 1.2 million photos into 1000 separate classes, with a top-5 test error rate of 6.7 percent, compared to 7.3 percent for the second-best entry. The enhanced usage of computing resources within the network is the fundamental feature of this architecture. Inception, a highly efficient deep CNN architecture, was deployed. Its core concept is based on how a convolutional vision network's optimal local sparse structure can be approximated and covered by conveniently available dense components.

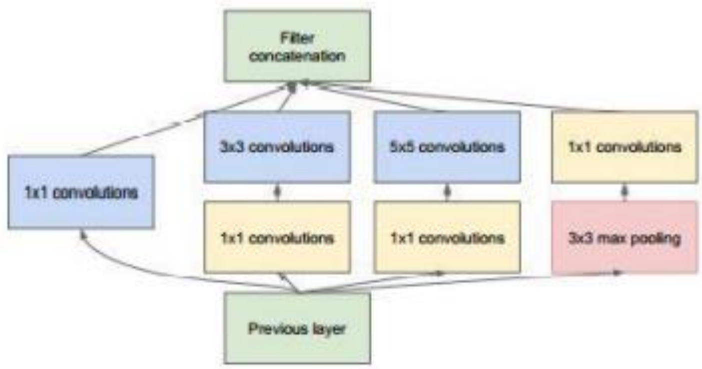

Figure 10: Inception module with dimension reduction Adapted from (Yenegeta, 2020)

Before the pricey 3 x 3 and 5 x 5 convolutions, the Inception module (as illustrated in figure 10) employed 1 x 1 convolutions to compute reductions. They are dual-purpose because, in addition to being utilized as reductions, they also contain the use of ReLU. There are 22 learnt layers and 5 pooling layers in GoogLeNet. It enhanced top-1 accuracy by around 0.6 percent by using average pooling instead of fully connected layers. ReLU non-linear activation is used in all convolutions, including those in the Inception module.

ResNet

ResNet took first place in the ILSVRC-2015 classification competition, which required it to classify 1.2 million photos into 1000 different classes with a top-5 test error rate of 3.57 percent. It also placed first in the ImageNet detection, ImageNet localization, COCO detection, and

COCO segmentation challenges. Despite the fact that ResNet is the deepest network ever published on ImageNet (with 152 layers), it is less sophisticated than VGG.

By adding a simple skip connection parallel to the layers of convolutional neural networks, deep residual networks (ResNets) make the training process faster and achieve higher accuracy than equivalent neural networks. Parallel to their normal convolutional layers, ResNets have shortcut connections. Shortcut connections skip one or more layers, and their main function is to perform an identity mapping without adding extra parameters or complexity, and their outputs are added to the stacked layers' outputs.

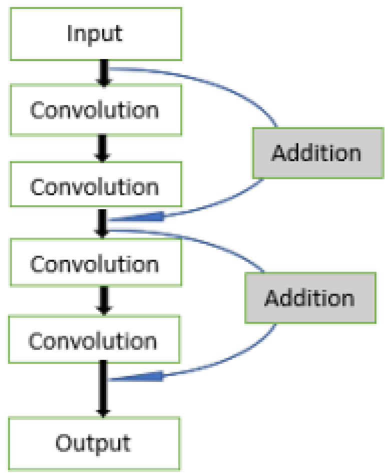

Figure 11: ResNet: a building block adapted from (Yenegeta, 2020)

$$Y = F (X, \{Wi\} + X. \quad (3.3)$$

$$Y = F (x, \{Wi\} + WsX. \quad (3.4)$$

As demonstrated in Figure 11, ResNets are defined as the serial connection of many basic blocks. In addition, each basic block has a shortcut link that is appended to its output. When the input and output dimensions are the same, identity shortcuts (Eqn. 3.3) can be utilized directly; otherwise, linear projection using shortcut connections (Eqn. 3.4) is employed to match the dimensions(Yenegeta, 2020).

3.2.4. Gabor Filter Feature extraction

Features are extracted directly from gray-scale character images by Gabor filters which are specially designed from statistical information of character structures. An adaptive sigmoid function is applied to the outputs of Gabor filters to achieve better performance on low-quality images. Gabor filters are widely employed in image processing and texture analysis because to their superior qualities, which include excellent joint spatial/spatial frequency localization and the ability to imitate the receptive fields of simple cells in the visual cortex(X. Wang et al., 2005).

Gabor filters are commonly used to represent and discriminate textures. To extract features, Gabor filters are used. The repetition of patterns over an image's surface is known as texture. The size, shape, and color of patterns vary from one image region to the next. Texture contains crucial information about the underlying structural arrangement of an image's surfaces. In texture feature extraction, Gabor feature extraction is effective(F. H. Wang et al., 2007)(Minhas & Javed, 2009).

The hand writer document features are extracted using 2D Gabor filters. Several kinds of Gabor filters are applied on photographs, with the form and orientation angles of the filters varied. Each document's results are analyzed and compared(Minhas & Javed, 2009).

As it is stated in (Minhas & Javed, 2009) 2D Gabor filter is defined as:

$$G(x, y; \theta, f) = \exp\left\{-\frac{1}{2}\left[\frac{x'^2}{\delta_x^2} + \frac{y'^2}{\delta_y^2}\right]\right\}\cos(2\pi f x')$$

$$x' = x\cos\theta + y\sin\theta \qquad (3.3)$$

$$y' = y\cos\theta - x\sin\theta$$

where δx and δy represent the spatial size of the filter, θ is the orientation angle, f is the frequency of the filter. Since a Gabor filter has both its real and imaginary parts, when an input image is convolved with a document of 20 filters, it yields 20 real and 20 imaginary outputs.

3.2.5. Support Vector Machine

With an ideal separation hyperplane, the Support Vector Machine (SVM) divides a set of input pattern vectors into two classes(Shih, 1369). Its goal is to separate the input patterns as accurately as possible by maximizing the distance between the closest vectors to the hyperplane. SVM creates the pattern classifier by using a variety of kernel functions as approximating functions, such as linear and polynomial functions.

SVM was created to address two-class classification problems, but it was later expanded to cover multi-class classification problems. Depending on the type of input patterns, different types of SVM classifiers are used: a linear maximal margin classifier is used for linearly separable data, a linear soft margin classifier is used for linearly non-separable, or overlapping, classes, and a nonlinear classifier is used for classes that are both overlapped and separated by nonlinear hyper planes.

Linear maximal marginal classifier: used where the training data can be separated by a hyperplane, w * x + b = 0. The goal of SVM is to find the optimal values for w and b. After finding the optimal separating hyperplane, w0. x + b0 = 0, an unseen pattern, xt, can be classified by the decision rule for f(x) = sign (w0. x + b0). Each unseen pattern, xi, belonging as it does to one of two classes, has a corresponding value Yi, where Yi = {-1, 1}. Because the hyperplane is w. x + b = 0, the training data can be divided into two classes such that:

w. xi + b >= 1, if Yi = 1

w. xi + b <= -1, if Yi = -1 --(3.4)

Linear soft margin classifier: handle input patterns that are overlapping or linearly non-separable. Its objective is to separate the two classes of training data with a minimal number of errors. The linearly separable case in the above case (linear maximal margin classifier), can be rewritten as:

w. xi + b >= 1 - ei, if yi = 1

w. xi + b <= -1 - e, if yi = -1------------------------------------ (3.5)

where ei, is non-negative slack variables.

Nonlinear classifier: When the input vectors cannot be linearly separated in the input space, nonlinear classifiers, such as polynomial functions, are employed to transform the input space to a higher-dimensional feature space.

3.3 Data Collection

This writer identification system for Amharic handwritten documents uses 2070 images of handwritten documents written in the Amharic language as input.

we gathered data from different organizations, coworkers, friends, and others with whom the researcher has a close relationship. We participate (70) peoples at an organization and individual level for the purpose of data collection. Some of them that is twenty-eight (28) of the writers are women, while the other one is that is forty-two (42) of the people are men.

As we have discussed in above chapter one; the participant peoples for this research are works on Amhara regional state police commission office, Amhara regional State court, Commercial Bank of Ethiopia workers especially; Belay Zeleke and Peda Branch's; Bahir Dar Preparatory School Grade 9,10,11,12 Students, and some of Voluntary guys are the participants for this study.

The researcher purposefully selected a document from Fana Broadcasting Corporate face-book address documents which is Posted in May 11/2013E.C. The reason to taken this document or writings was to make it free and neutral the writers from different perspectives or confusions and to make it formative and similar.

Then each writer is taken it ten sheets of A4 paper to write on. Individuals are requested to write the text over a period of time (more than 5 days) in which they were in various conditions in order to obtain a different handwriting.

The physical, psychological, and social status of writers are not taken into consideration while they are writing. There are almost 100 different Amharic characters and all numeric digits from 0 up to 9 in the selected text. Each of the handwritten document obtained from the writers is used for testing purpose by keeping the remaining twenty-eight documents as training data.

3.4 Preprocessing

Pre-processing is a common name for operations with images at the lowest level of abstraction. An image can be defined as a two-dimensional function, g (x, y), where x and y are spatial coordinates, and the amplitude of g at any pair of coordinates (x, y) is called the gray level or the intensity of the image at that coordinate or point.

The goal of the preprocessing stage is to improve image quality by removing undesired data from image content, such as noise, variation, and impractical features, as well as to increase image readability information for generating appropriate images. Image acquisition, image resizing, Noise removal, and Background removal or region of interest, are among the tasks performed during the preprocessing stage in this research.

3.4.1 Image Acquisition

Unless no additional processing is possible, image acquisition is the first stage in image processing. It refers to the process of taking an image with scanner hardware. Images can also be obtained through databases or other sources that are specifically designed for research. The majority of the time, a taken image is unprocessed and requires additional processing and analysis before it can be used for a specific purpose.

3.4.2 Digitization

Digitization is a process that converts handwritten hard-copy paper documents into a machine-readable format known as a document image, allowing for further processing. The developed system will thereafter be able to read the document image. The designed system then accepts several image formats. Individually handwritten paper for this study was scanned using an HP Scan Jet g4010 scanner with a 300-dpi resolution and stored in ‘jpeg' format.

A wide range of image file formats are supported by the Python Imaging Library. The library can recognize and read over 30 distinct file formats. Although write support is limited, it does handle the majority of common interchange and presentation formats.

Python commands can read, write, and display a variety of image graphics file formats. The common file formats of document image supported by python 2 are BMP (Microsoft Windows Bitmap), GIF (Graphics Interchange Files), JPEG (Joint Photographic Experts Group), PNG (Portable Network Graphics) and TIFF (Tagged Image File Format). All of the what we have discussed in the above are presented in the following Table.

Image formats supported in python3 are:

Format	Name	Supported Bit depths
‘bmp’	Windows Bitmap (BMP)	1, 4, 8, 16, 24, 32 bits
‘gif’	Graphics Interchange Files (GIF)	1, 2 to 8 bits
‘jpeg’	Joint Photographic Experts Group (JPEG)	8, 12, 16 bits
‘png’	Portable Network Graphics (PNG)	1, 2, 4, 8, 16, 24, 48 bits

‘tiff’	Tagged Image File Format (TIFF)	1, 8, 12, 16, 24, 32, 36, 48, 64 bits

Table 10: Image formats supported by python3

3.4.3 Resize Image

Image Size is the term given to describe the height and width of an image in pixels. As we know the image can be obtained either of the following two mechanism: i.e using camera or scanning of the hand writing document. For this paper we are using scanning to capture the image of handwritings of an individual. The reason Behind using scanner for image acquisition is well known; that is scanner has better resolution and bits-per-pixel color depth than a camera photo, it has also uniform lighting, no optical distortion of any sort, and it's perfectly flat than camera image. making reduction of images size plays an important role for the computation time of the algorithm.

Interpolation of original image pixels is the most popular resizing method. The three most widely used interpolation methods are nearest neighbor interpolation, bilinear interpolation, and bi-cubic interpolation. Bicubic interpolation algorithm can get relatively clear picture quality (Lin et al., 2014) and (Xia et al., 2013).

Bilinear interpolation: If the function's value is determined at the four corners, a bilinear interpolation strategy allows you to estimate the value of the function at any location in the rectangle's interior. The weighted average of the data at the four corners of the rectangle is used in bi-linear interpolation. The weights are calculated using the distance between the point and the corners for the (x,y) position inside the rectangle. The corners nearest to the pole are given more weight. Bi-linear uses 4 nearest neighbors to determine the output.

Bicubic Interpolation: as it is describing in the work of (Lin et al., 2014)For interpolating data points on a two-dimensional regular grid, bicubic interpolation is an extension of cubic interpolation. Surfaces produced by bilinear interpolation or nearest-neighbor interpolation are smoother than interpolated surfaces. Lagrange polynomials, cubic splines, and the cubic convolution technique can all be used to perform bicubic interpolation. When speed is not an issue, bicubic interpolation is generally preferred over bilinear or nearest-neighbor interpolation in image scaling. Unlike bilinear interpolation, which takes only four pixels (2x2) into

consideration, bicubic interpolation takes sixteen pixels into account (4x4). Images with bicubic interpolation are smoother and have less interpolation artifacts.
As per stated in the above, image resizing is mandatory for the computation time of an algorithm. Due to this reason the image pixel size from the scanner all most all lies between 800*280pixels in width and height respectively. In this work the images which comes from the scanner changed to 224*224 Pixel size in height and width to get appropriate computation time in related with the original image.

3.4.4 Noise Reduction

Digital capture of image can introduce noise (Unnecessary data) from scanning devices and transmission media which degrade the quality of document image(LEMMA, 1999; KASSA, 1998).
Smoothing (denoising or noise removal) operations are usually used to remove artifacts introduced during the image capture process. Neighboring stroke interference and underline are some of the problems that are often encountered(KASSA, 1998).
Unwanted black pixels and pen ink introduced into the document image through different conditions should generally not be part of the document are the noises of this investigation**.**
In order to remove noise from the image, different techniques have been proposed to deal with different types of noise depending on the application(MathWorks, 2002).
As it is stated in the book of(MathWorks, 2002) the followings are noise reduction methods:

1. **Linear Filtering**: To reduce specific types of noise, averaging or gaussian filters are utilized. The averaging filter can be used to remove grain noise from a shot. Local differences caused by grains are minimized since each pixel is adjusted to the average of the pixels in its area.
2. **The Median filter** is a statistically based nonlinear signal processing method. The neighborhood's median value replaces the noisy value of the digital image or sequence (mask). The mask's pixels are sorted by gray level and the group's median value is used to replace the noisy value. The median filtering output is

$$g(x,y) = med\{f(x-i, y-j), i, j \in W\}, \text{ where } f(x,y), \quad g(x,y) \quad \text{--------------- (3.6)}$$

are the original image and the output image respectively(Gao et al., 2010).

3. **Mathematician filter:** features a square measure that is designed to keep the time it takes to rise and fall to a minimum and to avoid overshooting. When collecting image frames from a

movie, a smoothing filter called a mathematician is one of twenty convolution techniques used to remove noise and blur(Dilpreet & Yadwinder, 2014).

4. **Wiener filter**: is used to reduce the amount of noise in a signal. Inverse filtering can be used to recover a picture that has been blurred by a known low pass filter. Inverse filtering, on the other hand, is extremely sensitive to additive noise. Wiener filtering achieves the best possible balance between inverse filtering and noise smoothing. It simultaneously removes additive noise and inverts the blurring. The Wiener filter reduces the difference in mean square error between the estimated and intended processes. In the process of inverse filtering and noise smoothing, it reduces the overall mean square error(Ibrahim et al., 2018).

5. **Gaussian filters** are commonly used to filter various types of surfaces. This type of filtration is the primary choice for filtration in many applications due to the algorithm's simplicity, ease of implementation, and robustness of the results. The linear Gaussian filter is frequently employed in surface characterization, has become an industry filtration standard, and has been widely used among researchers. By convolving the measured surface with a Gaussian weighting function, Gaussian filters can be applied to the input surface(khudayer Jadwa, 2018).

Segmentation

Segmenting images Classify pixels until objects or regions can be extracted from the background(Marius Bulacu et al., 2007). It is the process of separating the things of interest from the rest of the environment. The splitting of a picture into relevant parts in order to distinguish foreground from background is known as segmentation.

3.5 Feature Extraction

Feature extraction is a step in the dimensionality reduction process, which divides and reduces a large set of raw data into smaller groupings. As a result, processing will be simpler. The fact that these enormous data sets have a large number of variables is the most crucial feature. To process these variables, a large amount of computational power is required. As a result, feature extraction aids in the extraction of the best feature from large data sets by selecting and combining variables into features, effectively lowering the amount of data. These features are simple to use while still accurately and uniquely describing the real data set.

Feature extraction is the process of condensing a large number of existing features into a smaller number of new, more useful features. A lower number of more useful features are computed during feature extraction. It solves the problem of determining the most useful set of features.

Feature extraction is a technique for extracting descriptive features from images (which can include angle, shape, and orientation feature subcomponents). To understand relevant information from images, extracted features are used. The purpose of image feature extraction is to improve classification rates by extracting new features from raw pixel data to represent objects(Yenegeta, 2020).

Feature extraction and classification are the two essential aspects of a CNN. Several convolution layers are followed by max-pooling and an activation function in the feature extraction process. Typically, the classifier is made up of fully connected layers.

As we have discussed in chapter two; the most frequently utilized and efficient textural characteristics for offline writer identification are (Orientation, entropy and autocorrelation, Angle) as well as allograph features (graphemes) are used for this research, according to the review.

Orientation

This is a handwriting characteristic that describes how a writing stroke changes direction while composing handwritten text. For this feature, two edge fragments that emerge from the central pixel are considered in the neighborhood from the handwritten picture, and then the orientation for the joint probability distribution is computed for the two fragments(Marius Bulacu et al., 2003). It is believed that it will be more specific to the author, allowing for more precise identification.

Entropy

The Gray Level Co-occurrence Matrix extracts entropy as one of the features. It's a statistical measure of randomness that's used to describe the texture of an input image. The entropy measure utilized here focuses on the amount of information, which is normalized by the number of black pixels in the study areas. The following formula is used to calculate entropy.

$$E = -\sum \sum p\ i,j \log2\ i\ j\ p\ i,j \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.(3.7)$$

Where $p\ i,j$ is the i,j th entry in a normalized gray level co-occurrence matrix, it is a vector containing the probabilities of each pixel value in the image.

Autocorrelation

The normalized dot product between the original row and the shifted copy is computed after each row of the image is shifted onto itself by a certain offset. The total of all autocorrelation functions for all rows is then normalized to obtain a zero-lag correlation of 1. Regular vertical strokes will overlap in the original row and its horizontally shifted duplicate for offsets equal to integer multiples of the local wavelength, as detected by the autocorrelation function. As a result, the dot product contributes a significant amount to the final histogram.

The formula is: A= $\sum i \sum j\ i\ j\ p\ i,j$ …………………………………………… (3.8)

Gabor features, which are built from Gabor filter responses, have proved particularly successful in a variety of computer vision and image processing applications. For example, in biometrics, Doughman's iris code is the gold standard for iris recognition, while Gabor features are among the best performers in face recognition (e.g., fingerprint matching). Gabor features extract local data that is integrated to recognize an object or region of interest(Kamarainen, 2012).

3.6 Classification

Classification is the final and most significant step in the hand writer recognition process that results in the final output. The ultimate purpose of this stage is to compare a given unknown handwriting pattern with referenced handwriting patterns in order to assign one of the references for the unknown.

The process of classifying handwritten document images into known and specified writer classifications is referred to as classification. The next stage in offline writer identification is to categorize each image (or unit of handwritten image) to its respective writer class labels. The classification phase is responsible for accomplishing this. The time and effort put into carefully designing the strategies for the previous phases will help the classification step succeed.

The classification phase's ultimate purpose is to compare a given unknown handwriting document image (pattern) with known handwriting document images in order to assign one of the references to the unknown handwriting document. There are two main procedures to follow when identifying any pattern: training and testing. The classification algorithm builds a classifier or model in the first step, the training (also known as learning) phase, by analyzing a training set consisting of data object instances and their associated class labels, so that the classifier learns the association between training samples and their categories from the training dataset. The class category for a given unknown data tuple is predicted in the second phase, testing (also known as classification), using the model built in the previous step. The error analysis in the classification of unlabeled data, covered in the testing phase, is used to assess the classifier's performance. The error analysis in the classification of unlabeled data is described in the testing phase, which is used to evaluate the classifier's performance(Lundgren, 2016)(ABEBE, 2011).

Different classifier algorithms have been employed for different classification purposes. For this paper we use Convolutional Neural Network (CNN) and Support Vector Machine (SVM) are extensively used classifier algorithms for offline text independent writer identification systems; as it is stated in the work of (Kırlı & Gülmezo, 2012)(Gonzalez, 2007)

CHAPTER-FOUR

EXPERIMENT AND RESULT DISCUSSION

4.1 Introduction

This chapter describes in detail the experimental evaluation of the suggested model for automatic Hand writer recognition. The proposed model or architecture is approved after an experimental evaluation. The dataset used and the proposed model's implementation are thoroughly discussed.

4.2 Dataset

For this study, Hand writing of an individual images classified by CNN and SVM are taken from different individuals. There is no accessible dataset for academic research in Ethiopian handwriting recognition, or no ready-made data set for this type of research. So; We have taken images of the handwritings of an individual by using scanners. Because scanner has better optical resolution than camera and the size of the image when we are using scanner is A4 size or Appropriate size or standard size; whereas the image using camera is not standard size or A4 size.

As a result, we have prepared our dataset to train and test the proposed model's results. We gathered data for this study from Amhara regional state courts, Amhara regional police commission, Commercial Bank of Ethiopia Peda Branch, and Belay Zeleke Branch, Bahir Dar Preparatory school Grade 9, 10, 11, 12 both female and male students and some selected voluntary individuals. All written documents are taken in the form of images with JPEG (Joint Photographer Expert Group) format using scanjet4010 scanner with different pixel size; to make it similar pixel size for all images we are applying 224 x 224-pixel size of image resizing technique. Then we apply noise removal mechanism through median filter which is the most preferred for our study because of nature of noise which exists in our collected data. We also apply threshold segmentation or region of interest to be comfortable for feature extraction. Finally, we apply Feature extraction and classification techniques we describe below in detail. The data set we collected from the previous institution and individuals taken from Fana Broadcasting Corporate official web site Address on the day of May 11/2013E.C. The document that we need to rewrite shown below.

ጠቅላይ ዐቃቤ ህግ ከቪድን ለመከላከል ከወጣው መመሪያ ጋር በተያያዘ መሰረታዊ በሆኑ ጉዳዮች ላይ ማብራሪያ ሰጠ ስለ ስብሰባ የሚደነግገው አንቀጽ ውይይት ማድረግ የሚፈልግ የትኛውም ተቋም የተሳታፊዎቹን ቁጥር ከሃምሳ እንዳይበልጥ የሚከለክል ሲሆን ከሃምሳ በላይ ተሳታፊዎችን የሚያቅፉ ውይይቶች አስገዳጅ ሲሆኑ የአዳራሹን ወይም የመወያያውን ስፍራ ሰው የመያዝ አቅም አንድ አራተኛውን በመጠቀም እና ለሰላም ሚኒስቴር ወይም በተዋረድ ላሉ የክልል፣ ዞን እና ወረዳ የሰላምና ፀጥታ መዋቅሮች (በአቅራቢያ ላለ ፖሊስ ጣቢያ) በማሳወቅ ዉይይቱን ሊያከናውን ይችላል። ጠቅላይ ዐቃቤ ህግ ከቪድን ለመከላከል ከወጣው መመሪያ ጋር በተያያዘ መሰረታዊ በሆኑ ጉዳዮች ላይ ማብራሪያ ሰጠ ስለ ስብሰባ የሚደነግገው አንቀጽ ውይይት ማድረግ የሚፈልግ የትኛውም ተቋም የተሳታፊዎቹን ቁጥር ከሃምሳ እንዳይበልጥ የሚከለክል ሲሆን ከሃምሳ በላይ ተሳታፊዎችን የሚያቅፉ ውይይቶች አስገዳጅ ሲሆኑ የአዳራሹን ወይም የመወያያውን ስፍራ ሰው የመያዝ አቅም አንድ አራተኛውን በመጠቀም እና ለሰላም ሚኒስቴር ወይም በተዋረድ ላሉ የክልል፣ ዞን እና ወረዳ የሰላምና ፀጥታ መዋቅሮች (በአቅራቢያ ላለ ፖሊስ ጣቢያ) በማሳወቅ ዉይይቱን ሊያከናውን ይችላል።

Figure 12: The document written by hand writers for this study

Based on the above text document we collected data; that is there was 70 participants they come from the previous listed institution and voluntary individuals and we are asking to rewrite these document 30 times. Then the total prepared image dataset is 70*30*=2100 images prepared.

Accordingly, we have prepared our own data set that contains 2100 images and out of this we use 80% of the image that is 1680 images for training and 20% of the image for validation and testing purpose.

In the time of data collection, there are a lot of challenges have been occurred. Some of the previous listed organization employees were not have full willingness to rewrite the sample document in to the Given A4 paper. But some of them there are very eager to give as the document as per the schedule we agreed.

4.3 Experimental Result and Discussion

In experimental situations, a variety of factors influence execution time, including hardware attributes, dataset structure and complexity, and parallel job processing. Our testing was conducted on an HP laptop computer running Windows 10 with an Intel Core i7 processor and 8 GB of RAM, as well as the Google Collaborator cloud service. We used the Anaconda environment, which is an open-source offering for the Python programming language, to complete data preprocessing and paper preparation on the computer. The Python code was written in Jupiter Notebook. On the Google collab/cloud server, the feature extraction and classification tasks have been completed. We like to utilize Google Collaborator for feature extraction and classification activities since we can easily use any of the most recent Python packages, and because we are working with scanned image data, we can use GPU service if our data exceeds the CPU's capabilities. All of Google Collaborator's services are available for free. With the support of a Google collaborator, we used various Python packages both locally on the system and in the cloud. A list of Python packages is provided below.

TensorFlow and Kera's: they serve as environment for this paper simulation activities.

TensorFlow is an open-source library for numerical computation and large-scale machine learning that ease Google Brain TensorFlow**,** the process of acquiring data, training models, serving predictions, and refining future results.

Kera's is an open-source software library that provides a Python interface for artificial neural networks. Kera's acts as an interface for the TensorFlow library.

OpenCV (Open-Source Computer Vision Library) In the realm of computer vision, it is one of the most widely used libraries. OpenCV-Python is not just quick to code but also to deploy. As a result, it's an excellent choice for machine vision applications that require a lot of computation. In our model, OpenCV is employed for a variety of image processing functions.

Matplotlib: It is a cross-platform data analysis and graphical plotting package for Python and its numerical extension NumPy. It's most commonly used for 2D visualizations, although it can also handle photos. While Matplotlib can manipulate images and extract data from them, it does not support all file types.

NumPy: we use NumPy Because it supports arrays, it's one of the most important libraries in Python programming. A NumPy array of pixels represents data points in an image. As a result, we can manipulate an image's pixel values with simple NumPy operations like slicing, masking, and clever indexing. The image can be loaded with Skimage, and it can be displayed with Matplotlib.

Scikit-learn: Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines. And so, on

On the data set, the experiment for the study is thoroughly tested. The datasets that are used for testing are those that were discussed in the data collecting section. thirty experiments were conducted under each part, with the training and test datasets being varied. As previously stated, each handwritten image from the thirty handwritten documents of writers is used as a test sample, with the remaining twenty-nine images serving as training. Experiments carried out to test the effectiveness of our proposed model and described in detail.

4.3.1 Results on CNN

For this thesis report We use individuals handwriting image with an untrained CNN, which means that every pixel of every feature and every weight in every fully connected layer is set to a random value. Then we start feeding images through it, one after other.

Testing accuracy are achieved. Even though; if it is good achievement; still now there are a lot of works remaining in relating with CNN with SVM and Gabor filter. This finding was obtained using some data set which is collected from 70 individuals that is found on Amhara regional police commission office, Amhara regional state courts, commercial bank of Ethiopia Peda branch, and Belay Zeleke Branch, Bahir Dar Preparatory School and some selected voluntary individuals by asking to copy the given document 30 times repeatedly.

Precision, recall, and f1-score were utilized to assess our model's performance. In addition, for all of the abovementioned performance criteria, we determined the micro-average, macro-average, and weighted-average.

Convolution, activation function, pooling, and fully-connected layers comprise the training phase, with dropout occurring after the final fully-connected layers and before the SoftMax classifier. The model is trained using different parameters in the validation phase, such as batch normalization (after the activation function), and small dropout values at the beginning.

Our model, Hand writer recognition, achieves 94 percent training accuracy and 92.88 testing accuracy, by using CNN and default SoftMax Classifier as illustrated in Figure 13 below. And the total time to train the model is around 1 hours, taking on average 50 seconds per epoch.

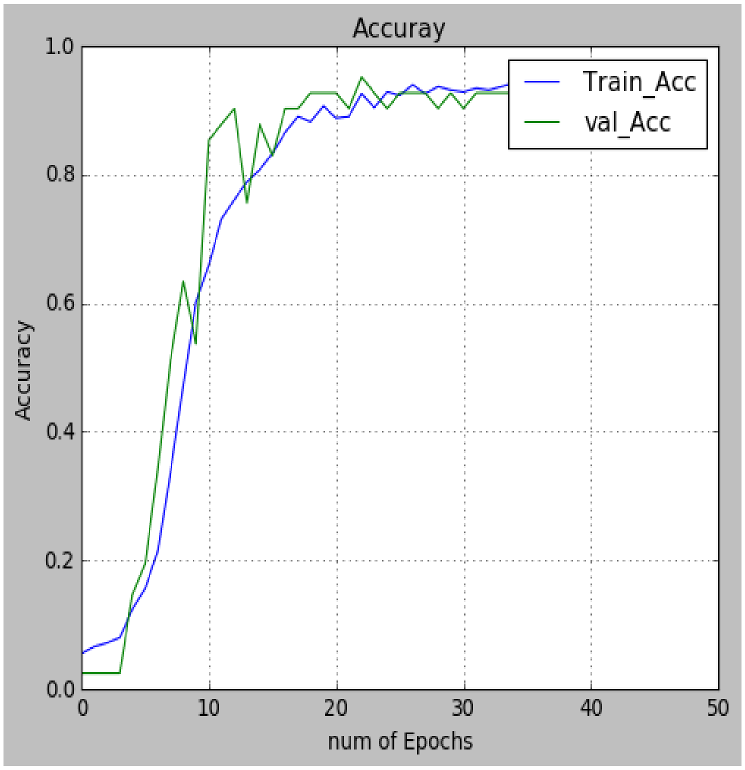

Figure 13: Training and validation accuracy curve of CNN model

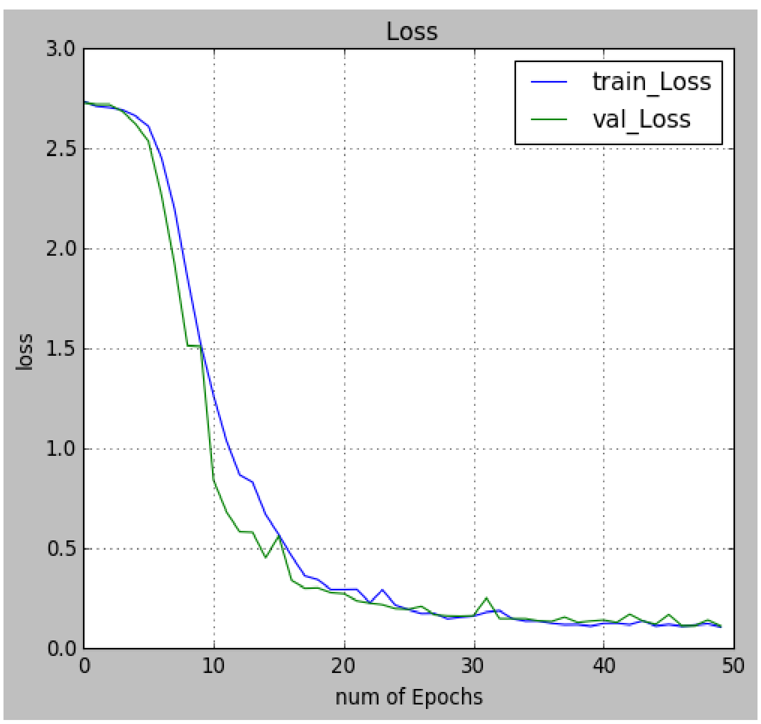

Figure 14: Training and validation loss curve of CNN model

4.3.2 Results on CNN combined-Gabor Filter with SVM

Our model, Hand writer identification System, has a training accuracy of 96 percent and a testing accuracy of 95 percent, as shown in figure 17 below. When the model is fed an image using a Gabor filter list and CNN, it achieves this classification accuracy. Instead of using simply CNN extracted image data, the model performs better when we provide an image filtered by the Gabor filter and CNN.

When the model is got it an image that has been extracted by the Convolutional neural network and Gabor Filter feature extraction, it achieves the above accuracy. The network improves by 2% from the training phase when we got it by using CNN Only.

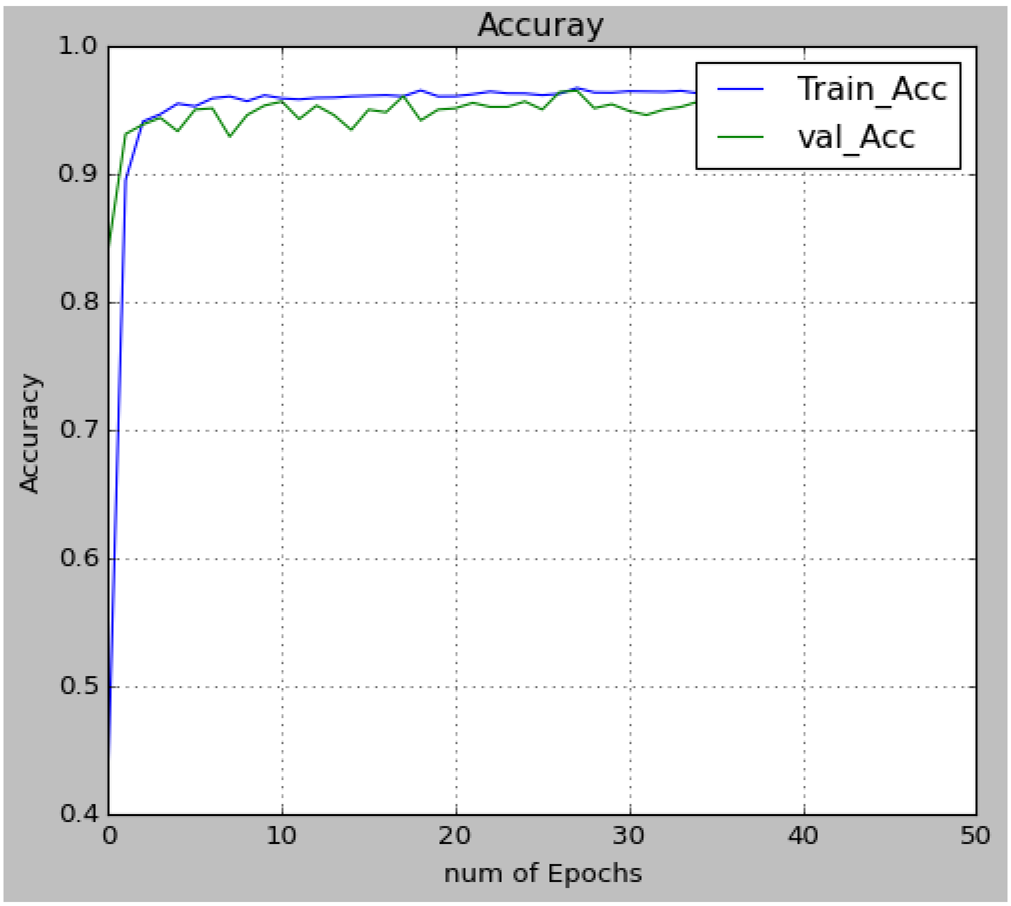

Figure 15: Model Training accuracy CNN-Gabor Filter with SVM Classifier

Training accuracy is increased nearly linearly and reaches 96%, while the validation accuracy goes a lot of up and down and reaches 95% accuracy at some instant.

The training and validation loss lowers as the number of epochs increases in parallel, as demonstrated in the training and validation loss curve in figure 16 below. Training loss is usually less than validation loss across the curve. Between training and validation loss, there is just a tiny gap. This suggests that our model is well-suited to our needs. In our models, there is no overfitting or underfitting. However, due to the similarity of the training and validation datasets, the training and validation losses are essentially identical at epochs 10 and 28.

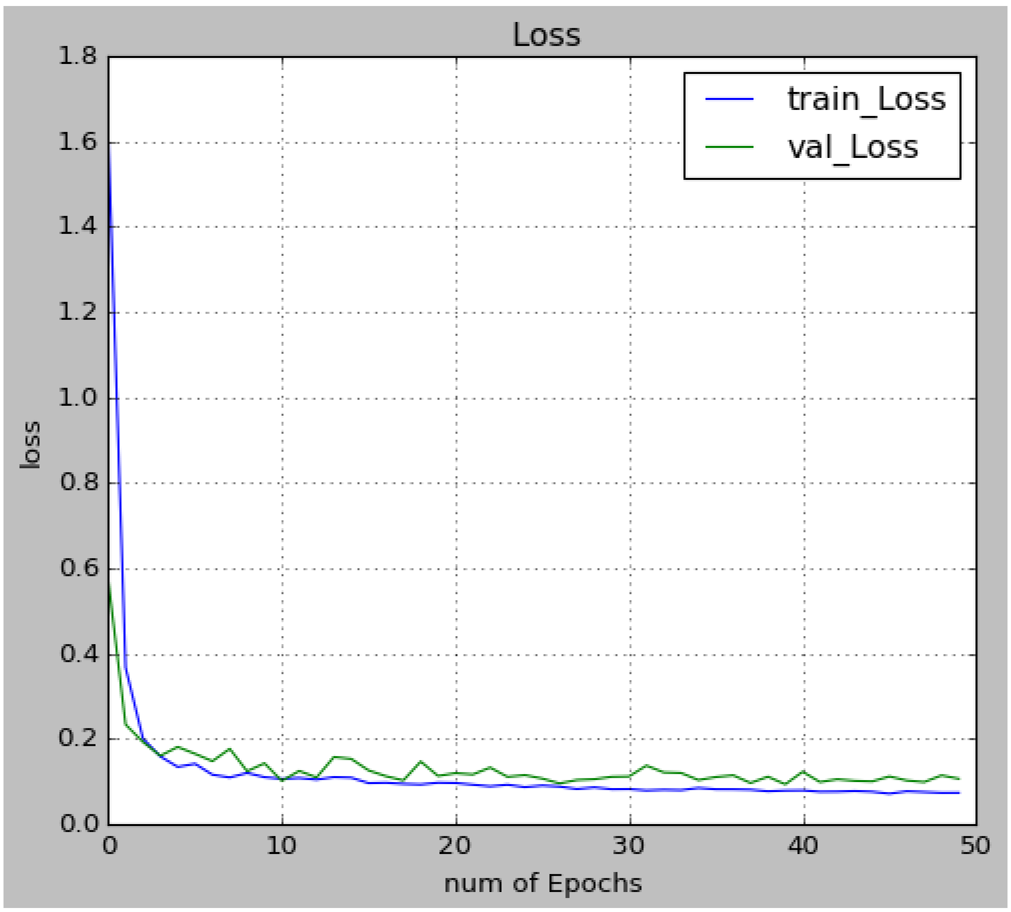

Figure 16:Training and validation loss curve of CNN-Gabor Filter with SVM Classifier

As we know for this study, we use 70 participant hand writers but we snapshoot the first 10 classes only because of many classes are there in the model. As shown from figure 17 below the testing accuracy of our model is 96%. The precision, recall, and f1-score of class 1 are 100%, 98%, and 99% respectively. Because we used deep learning CNN with Gabor Filter and SVM algorithms and many datasets that is why the testing or recognition accuracy increased. Based on the snapshoot below the remaining nine class's precision, recall, and f1-score are described on table 11 below.

	precision	recall	f1-score	support
Yalemsew	1.00	0.98	0.99	3
Alemnat	0.98	0.98	0.98	3
Wubedil	0.94	0.93	0.94	3
Zewudu	0.97	0.93	0.95	3
Amlakie	0.95	0.99	0.97	3
Girma	0.93	0.96	0.94	3
Debebe	0.95	0.95	0.95	3
Taye	0.97	1.00	0.99	3
Asya	0.98	0.91	0.95	3
Shimels	0.97	0.99	0.98	3
. . .				
accuracy			0.96	210
macro avg	0.96	0.96	0.96	210
weighted avg	0.96	0.96	0.96	210

Figure 17: Result of Accuracy, Precision, Recall, F1-score CNN combined with Gabor and SVM

As shown from figure 14 above result of Accuracy, Precision, Recall, F1-score and support (number of hand writers document support in each class in testing/recognition) CNN-Gabor Filter with SVM model are presented. Asya has relatively low recall accuracy because of dataset similarity (confusion) that are misclassified from its own class and classified from another class. The table 9 below shows the details of figure 14 above.

Hand writers	Writers Class	Precision	Recall	F1-score	Support
Yalemsew	0	1	98	99	3
Alemnat	1	98	98	98	3
Wubdil	2	94	93	94	3
Zewudu	3	97	93	95	3
Amlakie	4	95	99	97	3
Girma	5	93	96	94	3

Debebe	6	95	95	95	3
Taye	7	97	100	99	3
Asya	8	98	91	95	3
Shimels	9	97	99	98	3
	accuracy			0.96	210
	Macro avg	0.97	0.96	0.96	210
	Weighted avg	0.97	0.96	0.96	210

Table 11: Accuracy, Precision, Recall and F1-Score for CNN-Gabor Filter with SVM Classifier

4.3.3 Result Discussions

The training and validation accuracy and loss curves in figures 12 and 13 show that as the number of epochs increases, training and validation accuracy grow linearly, and training and validation loss decreases. Both the training and validation accuracy curves are more consistent (no up and down). Throughout the curve, training accuracy is nearly equal to validation accuracy, and training loss is nearly equal to validation loss. The difference between the training and validation accuracy curves is really small. Because our model fits our dataset, the problem of overfitting and underfitting is not seen in training and validation accuracy, as well as loss.

Because of the similarity of the validation and training datasets, validation and training accuracy at epoch 10 and 28 is practically identical.

We evaluated our model with 210 images, and the testing accuracy was 96 percent. The model's accuracy has improved as a result of the large number of datasets and the usage of preprocessing and a hybrid network for feature extraction and recognition. Performance Metrics, accuracy, precision, recall, and f1-score are used to test the model. Almost every student/participant in the class had the highest accuracy, precision, recall, and f1-score scores. This means that the CNN-Gabor Filter with SVM Classifier model performs the best. In the model of 210 test dataset, the confusion matrix was also employed to demonstrate correctly and incorrectly classified datasets in each class, as well as the overall. Because of the similarity of handwriting images, they are confusing each other, resulting in a dataset where one class is misclassified and classified to another. However, the character's shape, angle, orientation, and other characteristics no being alike to those of any other writer class; it is appropriately classified. That there is no misclassification. The table below shows the overall classification accuracy (correct and incorrect) as well as model performance.

Our model	Number of test image	Correctly classified	Incorrectly classified	Percent of correctly classified	Percent of incorrectly classified
CNN-Gabor Feature with SVM	210	201	9	96%	4%

Table 12:General correctly and incorrectly classified of CNN-Gabor with SVM Classifier model

4.3.4 Performance Comparison of Amharic Writer Identification with other Study

As previously stated, as far as in my searching there are only two papers done in related with Amharic hand writer recognition. Even if there are more works in other foreign language writer identification systems, the following is a comparison of this study with the two works that is Daniel and Eneyew's (Daniel Demoze and Eneyew Adugna, 2010) paper titled "Writer Identification for Ethiopic Handwriting." And (ALAMIREW, 2015)paper titled as “writer identification system for Amharic handwritten documents”.

Parameters	**(Daniel Demoze and Eneyew Adugna, 2010)**	**(L. ALAMIREW, 2015)**	**This study**
No. of writers/documents	25 writers, 50 documents	50 writers, 250 documents	70 writers, 2100 documents
Implementation tool	MATLAB	MATLAB	Python
Features extracted and used	Gabor Energy and Connected Component Contours	Grapheme, Edge hinge distribution, Autocorrelation and Entropy	Gabor filter, Entropy, shape, angle, and orientation
Classification algorithm/s used	KNN	ANN and KNN; separately and in combination	SVM

Evaluation metric/s used	Accuracy (percentage of correctly identified writers)	Accuracy and subjective evaluation	Accuracy, recall, precision, and f1 scores
Performance obtained (best results for minimum and maximum hit list sizes)	76% in hit list of size 1, using Gabor energy - 96% in hit list 3, using connected component	- 81.2%, hit list size of 1, Euclidean, Combination of ANN & KNN - 96.4 % hit list size of 10, Manhattan, Combination of ANN & KNN - Subjective evaluation - 82%	-CNN with Default Classifier SoftMax 94% -CNN-Gabor Filter with SVM Classifier 96% performance were achieved

Table 13: Comparison of this research with other pervious related study.

As it is described in the comparison of the above indicated table, Table 12, the result of this study is comparatively better than that of (Daniel Demoze and Eneyew Adugna, 2010) and (ALAMIREW, 2015) Some of the problems of their research were overcome by this research. The main differences of this research from (Daniel Demoze and Eneyew Adugna, 2010) and (ALAMIREW, 2015) research are summarized as follows.

- There are two aspects to (Daniel Demoze and Eneyew Adugna, 2010) research. And they conducted their research on the topic of writer identification utilizing those qualities individually. The study of (ALAMIREW, 2015)aimed to employ more features than they did, and to investigate the effect of fusing and mixing features in order to identify writers. However, this work uses a combination of CNN and Gabor filter features with an SVM classifier.
- The nearest neighbor classification approach was utilized in (Daniel Demoze and Eneyew Adugna, 2010)work. They didn't try to show us any other algorithms for identifying writers; In addition to the closest neighbor approach, The study of (ALAMIREW, 2015)used artificial neural networks on Amharic writer recognition systems, even if the results were not very effective. In this study we use CNN-Gabor Filter with Support vector machine in order to fix tinning problem. Diminishing

- For this study(ALAMIREW, 2015), developing a system for Amharic handwritten documents using a combination of edge hinge distribution and graphemes features over a combination of artificial neural network and nearest neighbor yielded a better result than (Tan et al., 2010) study, which used connected component contour and Gabor energy features separately on nearest neighbor algorithm. However, this study is better than the previous work in terms of building a system based on entropy, orientation, shape, and character angle.

CHAPTER-FIVE

CONCLUSION AND RECOMMENDATION

5.1. Conclusion

The conclusion of the suggested model for hand writer recognition for the Amharic language was discussed in this chapter. The study's major contributions and future work are also mentioned.

Now a day; identification of a given written document have a numerous advantage especially for the prevention of forgedity, crimes in different aspect and etc. on the other hand it has also a great advantage in the process of knowing historical document in historical places.

This study covers only the writers of Amharic language script only and proceeds a lots of processing steps to really identify the authors of the given document. As a conclusion; in this work we start ours task by preparing the document for rewriting purpose and then we asked the institutions and individuals to copied the sample prepared document in to a separate sheet of A4 paper; then we collected that document for preprocessing activities. And the we apply feature extraction of the image and apply classification strategy to assign relevant individual for the given document. Finally; we achieved 96% of accuracy.

5.2. Recommendation

Although the findings of this study are encouraging, there are several issues that need to be investigated further in order to improve performance and get it to an operational level in the future. As a result of this work, the researcher proposes the following issues for future research.

- The Amharic handwritten documents utilized in this study were gathered after people were asked to create a set of pre-written documents. We recommend to see the effect can be shown by looking at handwritten letters written by individuals for their own purposes like diary, memorandum and so on.

- As far as in our searching there has been no research on online writer identification of Amharic handwritten documents, therefore this is a sort of writer identification that can be done.
- As far as in our searching there has been no research on other Ethiopian languages except Amharic in related with writer identification of other Ethiopian language handwritten documents, therefore we recommend to be done in other Ethiopian languages in the future.

- The CNN with SVM classifiers classification algorithm was employed in this investigation. Other classification techniques, such as decision trees and other machine learning algorithms, should be tried with the Amharic writer identification system.

References

ABEBE, S. (ADDIS A. U. (2011). *BILINGUAL SCRIPT IDENTIFICATION FOR OPTICAL CHARACTER RECOGNITION OF MIXED AMHARIC AND ENGLISH PRINTED DOCUMENT*. *11*(2), 10–14.

Abushariah, A. A. M., Gunawan, T. S., Chebil, J., & Abushariah, M. A. M. (2012). Automatic person identification system using handwritten signatures. *2012 International Conference on Computer and Communication Engineering, ICCCE 2012, July*, 560–565. https://doi.org/10.1109/ICCCE.2012.6271249

Adam, G., & Josh, P. (2017). Deep Learning: A Practitioner's Approach. In *O'Reilly*.

Al-Maadeed, S. (2012). Text-dependent writer identification for arabic handwriting. *Journal of Electrical and Computer Engineering*, *2012*. https://doi.org/10.1155/2012/794106

ALAMIREW, L. A. (2015). *WRITER IDENTIFICATION SYSTEM FOR AMHARIC HANDWRITTEN DOCUMENTS*. 2–4.

Alder, M. (2001). *An Introduction to Pattern Recognition*. 561.

Ali, A. A. A., & Suresha, M. (2020). Survey on Segmentation and Recognition of Handwritten Arabic Script. In *SN Computer Science* (Vol. 1, Issue 4). Springer Singapore. https://doi.org/10.1007/s42979-020-00187-y

Anton, C., Stirbu, C., & Badea, R.-V. (2011). Identify Handwriting Individually Using Feed Forward Neural Networks. *International Journal of Intelligent Computing Research*, *2*(4), 169–174. https://doi.org/10.20533/ijicr.2042.4655.2011.0022

Ayalneh, W. (2020). *Online Handwriting Recognition of Ethiopic Characters Using Svm*.

Bharath Ramsundar, & Zadeh, R. B. (2017). TensorFlow for Deep Learning. In *Tf1* (Vol. 42, Issue 4).

Bulacu, M. (2007). Statistical Pattern Recognition for Automatic Writer Identification and Verification. *Ph.D. Dissertation, Dept. of Behav. and Soc. Sci., Univ. of Groningen, Netherlands*, 140.

Bulacu, M, & Schomaker, L. (2006). Combining Multiple Features for Text-Independent Writer Identification and Verification. *In Proc. of 10th IWFHR*, 281–286.

Bulacu, Marius, Schomaker, L., & Brink, A. (2007). Text-independent writer identification and verification on offline arabic handwriting. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, *2*, 769–773. https://doi.org/10.1109/ICDAR.2007.4377019

Bulacu, Marius, Schomaker, L., & Vuurpijl, L. (2003). Writer identification using edge-based directional features. *Proceedings of the International Conference on Document Analysis and Recognition*, *ICDAR*, *2003-Janua*(Icdar), 937–941. https://doi.org/10.1109/ICDAR.2003.1227797

Bulacu, Maruis, & Schomaker, L. (2007). Text-independent writer identification and verification using textural and allographic features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(4), 701–717. https://doi.org/10.1109/TPAMI.2007.1009

Chen, Q., Yan, Y., Deng, W., & Yuan, F. (2008). Handwriting identification based on constructing texture. *Proceedings - The 1st International Conference on Intelligent Networks and Intelligent Systems, ICINIS 2008*, 523–526. https://doi.org/10.1109/ICINIS.2008.159

Daniel Demoze and Eneyew Adugna. (2010). *WRITER IDENTIFICATION SYSTEM FOR ETHIOPIC HANDWRITING Daniel*. *27*.

Dilpreet, K., & Yadwinder, K. (2014). Various Image Segmentation Techniques: A Review. *International Journal of Computer Science and Mobile Computing*, *3*(5), 809–814.

Ding, S., Zhu, H., Jia, W., & Su, C. (2012). A survey on feature extraction for pattern recognition. *Artificial Intelligence Review*, *37*(3), 169–180. https://doi.org/10.1007/s10462-011-9225-y

Farnham, B., Tokyo, S., Boston, B., Sebastopol, F., Beijing, T., Hope, T., Resheff, Y. S., & Lieder, I. (2017). *Learning TensorFlow A Guide to Building Deep Learning Systems Revision History for the First Edition: Second Release*.

Fink, G., & Ploetz, T. (2006). Unsupervised Estimation of Writing Style Models for Improved Unconstrained Off-line Handwriting Recognition. *Proc. Int. Workshop on Frontiers in Handwriting Recognition (IWFHR), May*, 429--434.

Gao, W., Yang, L., Zhang, X., & Liu, H. (2010). An improved Sobel edge detection. *Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010*, *5*, 67–71. https://doi.org/10.1109/ICCSIT.2010.5563693

Gazzah, S., & Amara, N. Ben. (2008). Neural networks and support vector machines classifiers for writer identification using arabic script. *International Arab Journal of Information Technology*, *5*(1), 92–101.

Gonzalez, T. F. (2007). ImageNet Classification with Deep Convolutional Neural Networks. *Handbook of Approximation Algorithms and Metaheuristics*, 1–1432. https://doi.org/10.1201/9781420010749

Hellmuth, S. (2013). The Oxford handbook of Arabic linguistics. *The Oxford Handbook of Arabic Linguistics*, 45–70.

Ibrahim, N. B., Selim, M. M., & Zayed, H. H. (2018). An Automatic Arabic Sign Language Recognition System (ArSLRS). *Journal of King Saud University - Computer and Information Sciences*, *30*(4), 470–477. https://doi.org/10.1016/j.jksuci.2017.09.007

Kamarainen, J. K. (2012). Gabor features in image analysis. *2012 3rd International Conference on Image Processing Theory, Tools and Applications, IPTA 2012*, 13–14. https://doi.org/10.1109/IPTA.2012.6469502

KASSA, E. A. (1998). *Recognition of Formatted Amharic Text Using Optical Character Recognition (Ocr) Techniques*.

khudayer Jadwa, D. S. (2018). Wiener Filter based Medical Image De-noising. *International Journal of Science and Engineering Applications*, *7*(9), 318–323. https://doi.org/10.7753/ijsea0709.1014

Kim, I. J., & Kim, J. H. (2003). Statistical character structure modeling and its application to handwritten Chinese character recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(11), 1422–1436. https://doi.org/10.1109/TPAMI.2003.1240117

Kırlı, Ö., & Gülmezo, M. B. (2012). *Automatic writer identification from text line images*. 85–99. https://doi.org/10.1007/s10032-011-0161-9

LEMMA, D. T. (1999). *Optical Character Recognition of Typewritten Amharic Text*.

Leng, W. Y., & Shamsuddin, S. M. (2010). Writer identification for Chinese handwriting. *International Journal of Advances in Soft Computing and Its Applications*, *2*(2), 142–173.

Li, B., Sun, Z., & Tan, T. (2009). Hierarchical shape primitive features for online text-independent writer identification. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 986–990. https://doi.org/10.1109/ICDAR.2009.53

Lin, X., Ma, Y. lan, Ma, L. zhuang, & Zhang, R. ling. (2014). A survey for image resizing. *Journal of Zhejiang University: Science C*, *15*(9), 697–716. https://doi.org/10.1631/jzus.C1400102

Lundgren, S. (2016). Data Mining_ Concepts and Techniques. *Fight Against Idols*. https://doi.org/10.3726/978-3-653-01927-8/2

M, S., & Mary Idicula, S. (2011). A Survey on Writer Identification Schemes. *International Journal of Computer Applications*, *26*(2), 23–33. https://doi.org/10.5120/3075-4205

Marti, U. V., & Bunke, H. (2003). The IAM-database: An English sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, *5*(1), 39–46. https://doi.org/10.1007/s100320200071

Marti, U. V., Messerli, R., & Bunke, H. (2001). Writer identification using text line based features. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, *2001-Janua*, 101–105. https://doi.org/10.1109/ICDAR.2001.953763

MathWorks. (2002). *Image Processing Toolbox For Use with M ATLAB. Computation, Visualization, Programming.*

Mesay, H. (2003). *Department of Information Science Line Fitting To Amharic Ocr : the Case of Postal Address*. *July*.

Minhas, S., & Javed, M. Y. (2009). Iris feature extraction using gabor filter. *2009 International Conference on Emerging Technologies, ICET 2009*, 252–255. https://doi.org/10.1109/ICET.2009.5353166

Mishra, V. K., Kumar, S., & Shukla, N. (2017). Image Acquisition and Techniques to Perform Image

Acquisition. *SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology*, *9*(01), 21–24. https://doi.org/10.18090/samriddhi.v9i01.8333

Negussie, D. (2006). *Writer Independent Online Handwriting Recognition for Ethiopic Characters By Writer Independent Online Handwriting Recognition for Ethiopic Characters*. *June*.

Rehman, A., Naz, S., & Razzak, M. I. (2019). Writer identification using machine learning approaches: a comprehensive review. *Multimedia Tools and Applications*, *78*(8), 10889–10931. https://doi.org/10.1007/s11042-018-6577-1

Rosebrock, A. (2017). Deep Learning for Computer Vision with Python - Starter Bundle. *Pyimagesearch*, 330.

Said, H. E. S., Baker, K. D., & Tan, T. N. (2002). *Personal identification based on handwriting*. *33*, 1761–1764. https://doi.org/10.1109/icpr.1998.712068

Shih, F. Y. (1369). *Image Processing and Pattern Recognition*.

Shimeles, A. (2005). *Online Handwriting Recognition for Ethiopic Characters for Ethiopic Characters By Abnet Shimeles*. Angeles, L., Advocacy, S., Location, O. (2002).

Sulaiman, A., Omar, K., Nasrudin, M. F., & Arram, A. (2019). Length Independent Writer Identification Based on the Fusion of Deep and Hand-Crafted Descriptors. *IEEE Access*, *7*, 91772–91784. https://doi.org/10.1109/ACCESS.2019.2927286

Tan, J., Lai, J. H., Wang, C. D., & Feng, M. S. (2010). Off-line Chinese handwriting identification based on stroke shape and structure. *2nd International Conference on Information Engineering and Computer Science - Proceedings, ICIECS 2010*, *March*, 1–8. https://doi.org/10.1109/ICIECS.2010.5678229

Tan, J., Lai, J., & Zheng, W. S. (2013). Chinese handwritten writer identification based on structure features and extreme learning machine. *CEUR Workshop Proceedings*, *1022*, 21–25.

Theodoridis, S., Nos, K., & Bas, K. M. (2003). *Pattern recognition by Sergios theodoridis*.

Van Der Maaten, L., & Postma, E. (2005). Improving automatic writer identification. *Belgian/Netherlands Artificial Intelligence Conference*, *January*, 260–266.

Wang, F. H., Yao, X. H., & Han, J. Q. (2007). Iris recognition based on multichannel Gabor filtering and feature fusion. *Guangdian Gongcheng/Opto-Electronic Engineering*, *34*(12), 72–76.

Wang, X., Ding, X., & Liu, C. (2005). Gabor filters-based feature extraction for character recognition. *Pattern Recognition*, *38*(3), 369–379. https://doi.org/10.1016/j.patcog.2004.08.004

Wegman, R. C. (2010). The state of the art. *Renaissance? Perceptions of Continuity and Discontinuity in Europe*, *c.1300- c.1550*, *12*(8), 129–160. https://doi.org/10.1163/ej.9789004183346.i-370.21

Xia, P., Tahara, T., Kakue, T., Awatsuji, Y., Nishio, K., Ura, S., Kubota, T., & Matoba, O. (2013). Performance comparison of bilinear interpolation, bicubic interpolation, and B-spline interpolation in parallel phase-shifting digital holography. *Optical Review*, *20*(2), 193–197.

https://doi.org/10.1007/s10043-013-0033-2

Xu, L., Ding, X., Peng, L., & Li, X. (2011). An improved method based on weighted grid micro-structure feature for text-independent writer recognition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, *3*, 638–642. https://doi.org/10.1109/ICDAR.2011.134

Yan, Y., Chen, Q., Deng, W., & Yuan, F. (2009). Chinese handwriting identification based on stable spectral feature of texture images. *International Journal of Intelligent Engineering and Systems*, *2*(1), 17–22. https://doi.org/10.22266/ijies2009.0331.03

Yaregal, A. (2002). *School of Graduate Studies School of Information Studies for Africa Optical Character Recognition of Amharic Text : an Integrated Approach School of Graduate Studies School of Information Studies for Africa Optical Character Recognition of Amharic Text : June*.

Yenegeta, B. (2020). Detection and Grading of Trachoma Using Deep Convolutional Neural Network. *BMC Ophthalmology*.

Yoshua bengio. (2019). Deep learning. *Nature*, *29*(7553), 1–73.