
ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

# RESTORATION AND RETRIEVAL OF HISTORICAL AMHARIC DOCUMENT IMAGES

BIRUK MENGISTU

JUNE 2014

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

# RESTORATION AND RETRIEVAL OF HISTORICAL AMHARIC DOCUMENT IMAGES

A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information Science

By

BIRUK MENGISTU

JUNE 2014

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

# RESTORATION AND RETRIEVAL OF HISTORICAL AMHARIC DOCUMENT IMAGES

By

BIRUK MENGISTU

Name and signature of members of the examining Board

| **Name** | **Title** | **Signature** | **Date** |
|---|---|---|---|
| _____ | Chairperson | _____ | _____ |
| Dr. Million Meshesha (Ph.D) | Advisor | _____ | _____ |
| _____ | Examiner | _____ | _____ |

# DEDICATION

In memory of My Dad who was a Good Father and Advisor.

# Acknowledgment

Before all, I praise the almighty **GOD** and his mother **ST. MARY** for making everything the way it is. Next, I am so glad to express my warm gratitude to my advisor **Dr. Million Meshesha (PhD)** for being with me in all the ups and downs of this work. He is honestly the GREATEST teacher and advisor I have ever known. Thank you so much for showing and guiding me to an interesting research direction and recognizing my potential in addition to the care, support and valuable comments you gave me.

I would like to offer special thanks to all my family members specially **Seble Mengistu**, Genet Haile who encourage, love and support me with all you have for me.

Many thanks to Mr. Biniam Asnake, for providing me all the resources, the precious advices and supporting ideas you gave me. My sincere thanks also go to Amanuel Abate from National Bank Of Ethiopia who support me whenever I need help.

Last but not the least I like to thank peoples from Institute of Ethiopian Studies (IES) and National Archieve and Library Ageny (NALA) for providing me with valuable data for my thesis.

Biruk Mengistu

# Abstract

Many historical document image collections are now being scanned and made available over the Internet or in digital libraries. However, it is to be noted that effective access to such information sources is limited because of lack of efficient retrieval schemes.

The existing methods of searching and retrieving from document images can be conducted with the help of recognition-based (Optical Character Recognition) and recognition-free (Document Image Retrieval) or a combination of these two approaches. These algorithms try to analyze the global or local layout structure for different document images and estimate the similarity among them.

A few researches have been conducted to develop a recognition-free document image retrieval system that extracts information from document images relying on image features only. These systems are highly affected by degradation in historical documents which results from paper aging, folding or scanning. In this study, an attempt is made to integrate effective image restoring techniques to enhance the effectiveness of the system in searching within historical document images. This study also improves the online searching process of the system by accepting N-query terms for retrieving relevant documents in addition to image viewer, towards enhancing the interface to the Amharic Document Image Retrieval System.

In this study different images restoration techniques are experimented, such as Dilate, Erode and Combination of Mathematical Morphology techniques as well as Haar, Daubechies, and Symlet wavelet techniques. These techniques are experimented in historical documents as well as real life documents. Performance analysis shows that best result is obtained by combining mathematical morphology with Otsu thresholding. Finally, the performance of the system is evaluated before and after the integration of the selected restoring techniques in which an average overall performance of 87.02 % F-measure is registered in documents having low, medium and high levels of degradation with an improvement of retrieval effectiveness by 4.65 % F-measure. The performance registered in this study shows promising result for designing applicable Amharic document image retrieval. The major challenge is unavailability of standardized corpus and the dataset contains limited number of historical document images. Therefore, in the future a standardized corpus should be prepared and used for experimentation in similar studies.

# Table of Contents

# List of Tables

# List of Figures

# List of Equations

# List of Algorithms

# List of Sample Codes (Listings)

# List of Acronyms and Abbreviations

| | |
|---|---|
| ADIRS: | Amharic Document Image Retrieval System |
| DAR: | Document Analysis and Recognition |
| DIP: | Digital Image Processing |
| DIR: | Document Image Retrieval |
| DIRS: | Document Image Retrieval System |
| DTW: | Dynamic Time Warping |
| IES: | Institute of Ethiopian Studies |
| IR: | Information Retrieval |
| MATLAB: | MATrix LABoratory |
| MM : | Mathematical Morphology |
| MSE: | Mean Square Error |
| NALA : | National Achieve and Library Agency |
| OCR: | Optical Character Recognition |
| PSNR: | Peak signal to noise ratio |

# CHAPTER ONE

# INTRODUCTION

## 1.1 Background

In the past few years technological advancement provide the opportunity to surge of interest in digitizing documents such as books, articles and others to preserve them for posterity and ease of information extraction, retrieval etc [1].

Document image analysis has carved a niche out of the more general problem of computer vision because of its distinctness from regular class of images. Optical character recognition (OCR) was taken as one of the first clear applications of pattern recognition [2]. Even today, the challenges of complex content, noisy data, and use of new imaging devices keep the field active. It is increasingly becoming important to provide people with regular and effective access to the information [2]. Document images are information rich. Computer systems are used to develop digital technology systems, which enables easy access to the vast reservoir of information. These systems have an OCR at their core. Modern OCRs do not perform well in the case where the document image is substantially degraded [2]. Adequate enhancement approaches are required to make the document images fit for OCR. Further, the degraded images are not aesthetically appealing. These images are all departure from an ideal version of the document image, which is unambiguously well defined in the domain of machine-printed textual documents [2].

Searching and retrieval from document images can be conducted with the help of recognition-based (OCR), recognition-free (DIR) or a combination of these two approaches [6]. Optical Character Recognition (OCR) systems take scanned images of paper documents as input and automatically convert them into digital format for computer-aided data processing. Designing robust recognizers that are applicable for documents varying in quality, fonts, sizes and styles is known to be a long term solution [6][10].

OCR systems can be applied for the purpose of document image retrieval. However, the performance of the system relies heavily on the quality of the scanned images. Its performance deteriorates severely if the images are of poor quality or have complicated layout [74]. Other demerits of using OCR, according to Tan et al. [75], are requiring human correction, language dependence and waste of effort. A promising alternate direction is document image retrieval without explicit recognition or recognition-free approaches that search for relevant documents in the image domain [6].

An immediate need for effective access to digital libraries initiates document retrieval without explicit recognition. Recognition free document image retrieval (DIR) is designed basically using keyword spotting, where the document image is first segmented into words, and the user's keywords are located in the document image by word-to-word matching. Document image retrieval without explicit recognition becomes a very attractive field of research with the continuous growth of interest and requirements for the development of the modern society. It is a direct approach to access the digitized document image itself with certain level of performance. Especially for historical printed and handwritten document images, it is a promising approach to design an applicable retrieval system [6].

Processing historical document images are continuously receiving great attention by many researchers to restore and recover. The significant of the study enhance the quality of historical document and preserve properly for next generations as well as to make them accessible for public and research scholars such as historians, anthropologist, researchers and so on. Nowadays, producing digital copies of these documents to provide wider use of its rich resources while preserving rare, valuable and breakable documents[64]. Therefore, restoration of the degraded historical document images enables to remove these degraded effects and recover an image that is close to what one would be obtained under ideal printing and imaging conditions [2].

Digital images have great impact on our day-to-day life activities as well as on technology research areas [71]. The process of retrieving information from document image runs from digitization to searching for relevant documents based on users' query [19]. First, paper-based documents should be digitized using any of the image acquisition devices such as scanner and digital camera. After digitization, six major tasks are involved to retrieve information from document images. These are pre-processing, segmentation, feature extraction, indexing, matching and displaying relevant document images in ranked order. The overall structure of

document image retrieval system consists of two different parts: the offline and the online procedures [73]. Document images are often obtained by digitizing paper documents like books or manuscripts. They could be poor in appearance due to degradation of paper quality, spreading and flaking of ink toner, imaging artifacts etc [3]. All the above phenomena lead to different types of problems including boundary erosion, dilation, cuts/breaks and merges of characters. Further, with the advent of modern electronic gadgets like cellular phones, and digital cameras, the scope of document imaging has widened. As a result of this, document image analysis systems are becoming increasingly visible in everyday life [3].

The digitized image is the raw input to document analysis. The aim of the pre-processing module is to prepare the image for retrieval. Pre-processing involves binarization (or thresholding) and skew correction. It also undergoes image enhancements using restoration, noise filtering and increasing the contrast [6]. The great challenge faced with historical documents, such as the mixture of information (ink bleed through, spots), loss of information (gaps, discontinuities), pattern deformation (geometric deformation, noise around contours, unwanted jagged edges), appear leading to apply restoration techniques [64].

Then, the image is segmented to separate the set of words in a document. Segmentation occurs at two levels. On the first level, text, graphics and other parts are separated. On the second level, text lines and words in the image are located [6]. Image segmentation leads to more compact image representations by partitioning an image into a set of disjoint words to represent document images [9].

Segmentation is followed by feature extraction, which involves extracting the meaningful information from the document images [6] [9]. Based on feature vectors, document images are indexed to speed up searching from a document collection. Indexing is an offline process that enables to organize document images using extracted features of word images [9].

During searching the system accepts query from users and then the text query is converted to image (by a process called query rendering) in order to be compared with set of document images. To represent query word images, an image feature based on the pixel values in the predefined area is extracted. The idea is to calculate vector value for the image and represent each word image in one vector [8]. Then, matching in document images can identify the word

images of the documents that are more similar to the query word through the extracted feature vectors [9]. Similarity measurement is a central problem in computer vision and pattern recognition, which has wider applications in multimedia retrieval, especially in content based image retrieval [6].

The search results are finally presented to the user in ranked order. The main aim of ranking is to sort the retrieved documents according to their degree of relevance to the query provided by the user. Consequently, ranking algorithm is at the core of document image retrieval and operates according to basic premises of document relevance in which different set of premises yield distinct document retrieval models. Finally, performance evaluation of information retrieval systems is important to measure the efficiency and effectiveness of the retrieval process [6] [9] and identify further research areas.

## 1.2 Statement of the problem

Amharic is an official working language of the federal government and most regional states of Ethiopia, with more than 20 million speakers [4]. Amharic is today probably one of the five languages with largest speaker on the continent [5]. Accordingly, there are huge amount of hardcopy Amharic documents that are available in government and non-government offices. The language has its own script called FIDEL using which a number of documents are published in the form of books, magazine, newspapers, etc [6].

In an effort to digitize such documents there is a need to design and develop document image retrieval system that ease searching for the relevant documents. To this ends there are attempts on the implementation of document image retrieval system to ease searching in document images written in Amharic language.

Million [6] explores novel approaches for understanding and accessing the content of document image collections that vary in quality and printing and propose the need to apply advanced image pre-processing techniques for document analysis in-order to enhance the performance of DIR system.

Mesfin [7] designed a document image retrieval system that provides response to user's query without the requirement of character recognition. As continuation of Mesfin's [7] work, Abreham [8] designed a retrieval system that accepts user query and searches from Amharic document image corpus. Adane [9] explored feature extraction and matching techniques that are crucial to enhance the performance of the document image retrieval system. Adane also indicates that Pre-processing techniques like image restoring helps to increase the effectiveness and efficiency of the system.

Further, Biniam [l0] attempts to improve the effectiveness of the system by integrating noise removal scheme and adopting an information retrieval (IR) model. However, he boldly indicates that in order to come up with a practical Amharic document image retrieval system, image pre-processing tasks such as image restoration and edge detection are vital especially for real-life and historical documents. Finally, Gedion [12] worked on Page segmentation and Kibrom [11] worked on word segmentation from historical Amharic document images having different outputs. The main challenge faced by Kibrom was that the document images he used are affected by degradation and he recommended applying a proper restoration techniques in-order to improve ADIRS.

Most of the historical Amharic documents are poor in quality and need to apply restoration techniques. As to the researcher's knowledge there is no study that explores restoration of on Amharic historical document images. Therefore, the main concerns of this study is to explore the application of different document images restoration techniques to improve the quality of historical Amharic document images for enhancing document image retrieval for Amharic language.

To this end, this research attempts to explore solutions for the following research questions

- To what extent historical Amharic documents are degraded?
- What technique is suitable for restoring the quality of historical document images?
- To what extent restoring degraded documents enhance the performance of Amharic document image retrieval system?
- How to enable vector space model handle N-words queries in image space?

- How to design a user friendly interface that can support retrieved document image visualization?

## 1.3 Objective of the study

This research work has a general objective and a list of specific objectives in order to solve the problems that initiate this study.

### 1.3.1 General Objective

The general objective of this research is to enhance the performance of Amharic document image retrieval in searching for relevant historical document.

### 1.3.2 Specific Objectives

In-order to achieve the general objective, the study accomplishes the following specific objectives.

- To review literature on previously related works for conceptual understanding and then identify different techniques and algorithms for document restoration.
- To identify suitable techniques for Amharic document images restoration.
- To study the nature and type of degradation on Amharic document image and design efficient restoring technique.
- To adopt restoration technique for Amharic historical document images and integrate with the Amharic document information retrieval system.
- To introduce a classic vector space model that enables to search N-words query accepted from the user.
- To improve usability of the interface and image viewer of ADIRS.
- To evaluate effectiveness of the information retrieval system using evaluation techniques such as recall, precision and F-measure.

## 1.4 Scope and Limitation of the study

This research is a continuation of previous researchers [7][8][9][10][11] on developing Amharic document image retrieval system. It adopts Amharic document images restoration techniques and integrates with the document image retrieval system. This study also enhances usability of the interface of ADIRS, as well as the functionality of vector space model. The retrieval scheme considers different historical Amharic documents images found in different libraries.

For performance measure, the document images are categorized into three levels of degradation (low, medium and high) and for testing data corpus was collected from different sources such as historical letters written in qum tshefet, old history books, and bibles.

The limitation of this study is that, due to unavailability of standardized corpus and time limitation, the dataset contains limited number of historical document images for experimentation.

## 1.5 Methodology of the study

Methodology is a way to systematically solve the research problem. It may be understood as a science of studying how research is done scientifically [12]. Accordingly, this research followed experimental research since it involves dataset preparation, system development and evaluation.

### 1.5.1 Literature Review

Related literature from different sources such as books, journals, articles, conference proceedings and the Internet are reviewed to understand in detail the field of document image retrieval, to select tools and procedures suitable for developing the system particularly in restoring the quality of degraded Amharic document images. Furthermore, additional reviews are conducted in-order to cope up with the previous researches on Amharic document image retrieval system.

### 1.5.2 Dataset Preparation

To investigate the performance of the proposed system the necessary historical Amharic document images from different Amharic historical documents collection called "Qum Tsehfet" from Institute of Ethiopian Studies (**IES**), different historical letters written collection for Ras Teferi former king of Ethiopia and old written histories from National Achieve and Library Agency (**NALA**) were collected. A camera with a resolution of 300 dots-per-inch (DPI) is used. 300 DPI is the preferred and efficient level for historical documents because it does not tend to break thin lines or fill gaps [49].

To evaluate the performance of the system, 2987 word images (1145 from low level degradation, 992 from medium level degradation and 850 from high level degradation) are collected. They are used for testing the performance of the restoring algorithm before and after integrating with the current Amharic document image retrieval system.

### 1.5.3 Implementation Tools

Advanced Java image-programming language and MATLAB are used. The reason why the researcher chose Java programming language and MATLAB is, first, this research is a continuation of previous work of Mesfin [7], Abreham [8], Adane [9], Biniam [10] and Kibrom [11] which was implemented using Java Programming Language. The other reason is familiarity of the researcher with Java Programming Language and MATLAB. Java has a number of built in packages which helps for image processing and also supports to write modules source code which is reusable and easily modifiable [13].

MATLAB is a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numeric computation and it also used in a wide range of applications, including signal and image processing, communications, control design, test and measurement, financial modeling and analysis, and computational biology [50].

MATLAB code can be integrated easily with other languages and applications [51]. Accordingly, after developing the image restoration module in MATLAB, we used *MATLAB Builder JA* software to easily and successfully integrate the prototype of the proposed system with the previous works.

### 1.5.4 Evaluation techniques

Six image restorations algorithms were explored in order to come up with restored documents. These are Erosion, Dilation and Combination of morphological operation from mathematical morphology and Haar, Dabechies and Symlet algorithms from wavelet transformation. The performance of the algorithms is measured using the common image quality measurement of Peak Signal to Noise Ratio (PSNR) which is calculated based on the Mean Square Error (MSE). The MSE represents the cumulative squared error between the compressed and the original image, whereas PSNR represents a measure of the peak error. The PSNR block computes the peak signal-to-noise ratio, in decibel (dB), between two images. This ratio is often used as a quality measurement and the higher the PSNR, the better the quality of the reconstructed image [9][14].

Finally, the best performer image restoration technique is selected and integrated with the previous system where the retrieval effectiveness was measured using the three most widely used measures: recall, precision and F-measure. Recall evaluates the ability of the system to retrieve relevant documents from total number of relevant documents, where precision evaluates the ability of the system to retrieve only the relevant documents from the retrieved documents [14]. Finally, F-measure is used to analyze the maximum recall and precision that is registered in this study [14].

## 1.6 Significance of the study

Different documents articulated and printed in Amharic are piled high in information centers, libraries, museums, and government and private organizations [8]. This research work was attempt to integrate by restoring degraded historical Amharic documents in addition to searching with N-words query, image viewer and enhancing the interface of ADIRS to the previously developed Amharic document image retrieval system [7] [8] [9] [10] [11]. The results of the study have a significant contribution to preserve and properly utilize the knowledge embedded in the documents as well as to permit easy access to these document images by different organizations.

In addition to that, the restoration techniques can be applied to other related Ethio-semantic language groups. Furthermore, it will provide a leap forward to realize the dream of developing full-fledged, effective and efficient Amharic document image retrieval system (ADIRS).

## 1.7 Organization of the study

This thesis is organized into five chapters. The first chapter discusses the background of the study and statement of the problem. It also presents general and specific objectives of the study, methodology of the study, scope and limitation of the research and application of the investigated results.

In chapter two, literature review on information retrieval, document image processing and retrieval is presented. Moreover, a brief review of the Amharic language, development and characteristics of the Amharic writing system and their challenges, international and local related works on document image retrieval are discussed.

Chapter three describes the proposed image pre-processing techniques and algorithms. Image restoration techniques are briefly discussed with their equation and algorithms. The evaluation measures that are used for measuring the performance of each image restoration algorithms are also presented.

Chapter four emphasizes the integration of image restoration techniques and experimental results that are used to confirm the validity of the proposed image pre-processing techniques to retrieve relevant information's from degraded Amharic document images.

Finally, based on the findings of the study, conclusion and recommendations of the research are forwarded in chapter five.

# CHAPTER TWO

# LITERATURE REVIEW

The present growth of digitization of documents demands an immediate solution to enable the archived valuable materials searchable and usable by users. This requires research in the area of document understanding, specifically in the area of document image retrieval [6].

## 2.1 Information retrieval

Information retrieval (IR) deals with the representation, storage, organization of, and access to information items [34]. The representation and organization of the information items should provide the user with easy access to the information in which they are interested. It responds to the user queries that can be words from a natural language or it can be multimedia information. The similarity between the content of the document and the query determine the relevance of retrieved information [15].

Information retrieval systems retrieve documents from the collection, based on their representation formats. These documents are retrieved following step by step procedures like query representation of user need, matching and then ranking based on likelihood of relevance and finally, present the documents to the user [15]

In the past 20 years, the area of information retrieval has grown well beyond its primary goals of indexing text and searching for useful documents in a collection [34]. Nowadays, research in **IR** includes modeling, document classification and categorization, systems architecture, user interfaces, data visualization, information filtering, language understanding, language identification and multimedia retrieval including document image retrieval [15].

## 2.2 Document image retrieval

Digitization of documents demands an immediate solution to access them electronically. This will enable the archived valuable materials to be searchable and usable by users in order to achieve their objectives. As these improvements continue, document image analysis systems will become increasingly more evident in every-day life and this requires research in the area of document image understanding, specifically in the area of document image recognition as well as document image retrieval [6].

Information retrieval in document image databases has become a growing and challenging problem. This is because of the diversity in quality of the documents that are digitized and available for use. Additional challenge in searching for relevant document images using query words is the existence of word-form variants (they vary in quality, scripts, fonts, sizes and styles). This requires better document image understanding schemes to make the content of these documents accessible to users through indexing and retrieval of relevant documents. Thus, searching and retrieval for relevant document images can be done with the help of the following approaches: Recognition-based, Recognition-free or sometimes combination of the two [6].

### 2.2.1 Recognition based retrieval

Recognition based retrieval is the usual way of information retrieval from scanned document images. This recognition technique digitize document images and then perform optical character recognition(OCR) to transform characters, which were contained in the digitized printed document into a machine-editable text (character-coded text) such as in the American Standard Code for Information Interchange (ASCII) or Unicode format [16,17,18].

Optical Character Recognition is the process of examining printed or handwritten characters on paper and determining their shapes by detecting patterns of dark and light. Once the scanner or reader has determined the shapes, character recognition methods and pattern matching with stored sets of characters are used to translate the shapes into computer text or computer readable format [19]. There are two type of system with respect to the way the input is provided to the

system. The first being online Optical Character Recognition where the recognition task is performed concurrently with the writing process. The other one is offline Optical Character Recognition process which performed after the whole text is scanned and bit map representation of a text is supplied to the system [19].

### 2.2.2 Recognition free retrieval

The recognition-based approach has some limitations when dealing with documents having a high level of noise/degradation or containing multi-lingual text printed with non-standard fonts with a variable layout .To overcome such problems a promising direction is to search for relevant documents using only image properties, without explicit recognition [21]. Consequently, document image retrieval without explicit recognition becomes a very attractive field of research with the continuous growth of interest and requirements for the development of the modern society. As a result, document image retrieval without explicit recognition is proposed as a short-term solution[20,23].

## 2.3 Steps in document image retrieval

In many businesses today, imaging systems are being used to store images of pages to make storage and retrieve more efficient [22]. As a result, document images have become a popular information source in our modern society, and information retrieval in document image corpus is an important topic in knowledge and data engineering research.

There are a number of steps followed in developing retrieval systems for document images to speed up searching from a document image collection. Figure 2.1 presents the general architecture for developing document image retrieval system without explicit recognition [6][10].

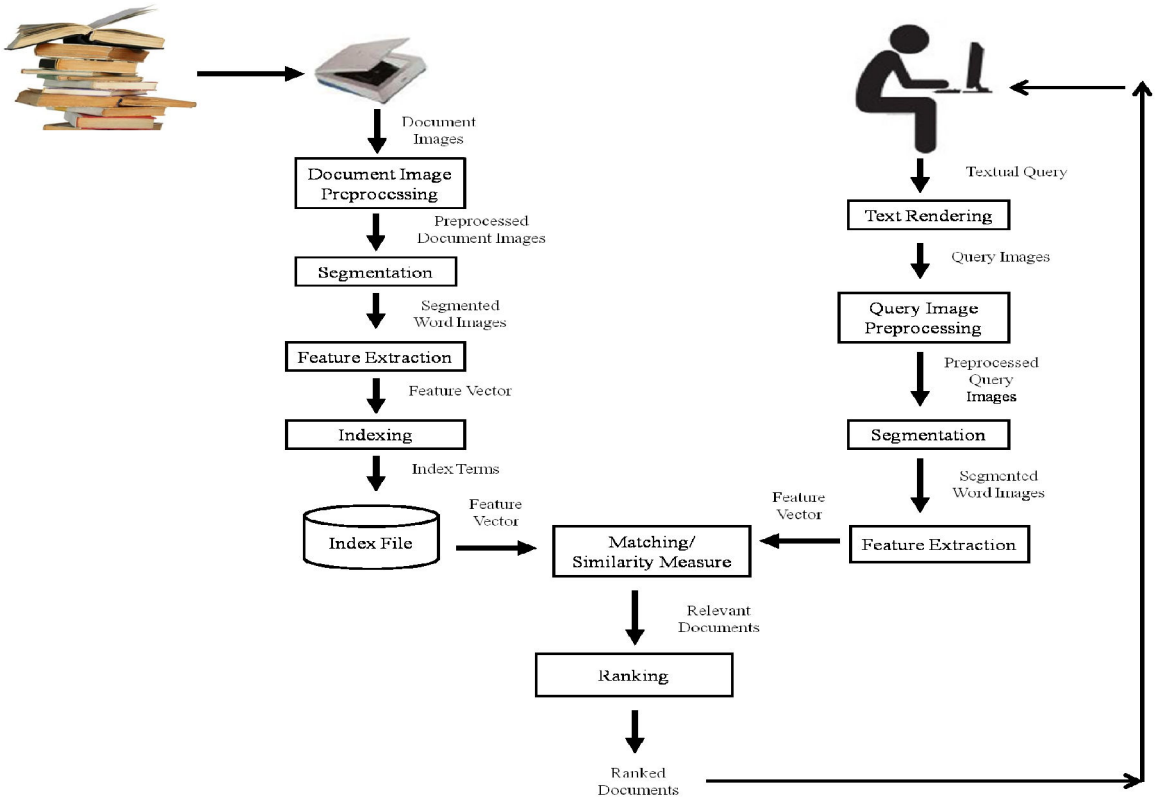



**Figure 2.1: The overall architecture of the Document Image Retrieval System (DIRS)**

The first task that must be done in order to manipulate and process images is digitization. Image acquisition is the process of acquiring or obtaining the image of document in color, gray level or binary format. A digitizer such as camera or scanner converts hardcopy documents to digital data [79]. After digitization, the next vital process in document analysis is to perform pre-processing on this image to prepare it for further analysis. The pre-processed images are then passed through the Segmentation, feature extraction, indexing, matching and ranking phases for developing document image retrieval system. Segmentation is the process of identifying the objects of our interest. Segmentation occurs on two levels. On the first level, if the document contains both text and graphics, these are separated for subsequent processing by different methods. On the second level, segmentation is performed on text by locating columns, lines and finally words [28].

Features are a representation of objects like word images. Features should also be distinct with respect to their neighbourhood, stable with respect to noise, and complementary with respect to other features [30]. Feature extraction involves the identification of the meaningful information from the document image so that it reduces the storage required [79]. Feature extraction is the

problem of gathering information from raw data, which is most relevant for a given application [6]. The features that are extracted from whole image are known as the global features and the features that are extracted from blocks identified during segmentation or from subdivision (sub-sectioning) of the document are known as local features [79].

It is common in IR to represent each document by means of keywords or index terms. These are usually derived from the text or some surrogate (e.g abstract) through a process of *indexing*. In addition to the selection of terms to represent documents, it is common to also associate weights that reflect the importance of each term as an indicator of the content of the documents to which it is assigned. Thus, in designing search strategies, it is reasonable to consider a document-by - term matrix as the information one starts with, where the (i, r)$^{th}$ element of the matrix corresponds to the weight of term "i" in document "r"[34]. In what follows, we denote this matrix by D, having elements $d_{ir}$s.

Given the matrix D and our desire to rank documents, there are several different ways to model the search problem. One approach which has been widely used over the years is to model documents and queries as a vector [9][33]. Here, $d_{ir}$s is considered to be i$^{th}$ component of the vector representing the r$^{th}$ document. When a query is presented, the system formulates the query vectors based on a chosen method of determining similarity between vectors. For example, similarity between the query and the document may be defined as the scalar product of corresponding vectors and the documents could be ranked in decreasing order of their similarity.

In retrieving information from document images using a word query, ranking the retrieved documents is possible based on similarity measure from queries weights [8]. In other way of saying that computing dissimilarity between the retrieved document images and the query image is used to rank documents according to their relevance. The dissimilarity values do not represent the degree of dissimilarities of the retrieved images with respect to the query; rather provide a means to rank the retrieved images [33].

## 2.4 Document image restoration

Images of paper documents are almost inevitably degraded in the course of printing, photocopying, Faxing, and scanning, and this loss of quality - even when it appears negligible to human eyes - can be responsible for an abrupt decline in accuracy by the current generation of

text recognition (OCR) systems. This fragility of OCR systems when confronted by low image quality is well known to the OCR community [37]. The accuracy of today's document recognition algorithms falls abruptly when image quality degrades even slightly [35]. The physical causes of image degradation are myriad: spreading and flaking of ink toner, uneven paper surface, low print contrast, non-uniform illumination, defocusing, finite spatial sampling rate, variations in pixel sensor sensitivity and placement, noise in electronic components, binarization (e.g. fixed and adaptive thresholding) and, images may result from more than one stage of printing and imaging[35].

Document images are scanned from pseudo binary hardcopy paper manuscripts with a flatbed, sheet-fed, or mounted imaging devices. Recently, however, the community has seen an increased interest in adapting modern imaging device like digital cameras to tasks related to document image analysis [36]. Although they cannot replace scanners, they are small, light, easily integrated with various networks, and more suitable for many document capturing tasks in less constrained environments. These advantages lead to a natural extension of the document processing community where scanners are used to image hardcopy documents or natural scenes containing textual content. This has given rise to new potential applications, though most of the time handicapped by low-resolution. For text and document analysis, as the application areas extend to lower resolution capturing devices, document image restoration methods are becoming more important and necessary[36].

## 2.5 Type of degradation

Degradation can be seen as a wide variety of less-than-ideal properties of real document images, for example coarsening due to low digitizing resolution, ink/toner drop-outs and smears, thinning and thickening, geometric deformations. [38].

A document can appear degraded in multiple ways. The reasons for the degradation may vary from poor source type and the image acquisition process to the storage environment that directly causes problems for the image quality. Degradation is unquestionably one of the main reasons for image processing to fail. Most degradation types in document images affect both physical

and semantic understandability in the document analysis tasks, such as segmentation, feature extraction classification and optical character recognition.

A typology for different types of degradation on old document images has been proposed by Drira [39]. This typology was preceded by an exhaustive research of all degradations, which was done by consulting various images of degraded documents. The proposed classification was made according to the further treatment that will be applied in the context of virtual document image restoration. It is decomposed into three classes [39]: **background degradation, foreground degradation and global degradation.**

### 2.5.1 Background Degradation

The most common degradation is characterized by the presence of artifacts in the background of documents. It includes water blotches due to humidity, marks resulting from ink that traverses the paper (bleed-through) and from the scanning process (show-through). Ancient historical documents generally comprise a combination of these degradations [40].

For example: **Water blotches** (see Fig. 2.2) are characterized by having a mainly convex shape (due to the diffusion of water molecules in the paper), a color that is darker than the neighbourhood (due to the dust which is attracted in the paper texture), and an even darker border area where the dust accumulates [40].

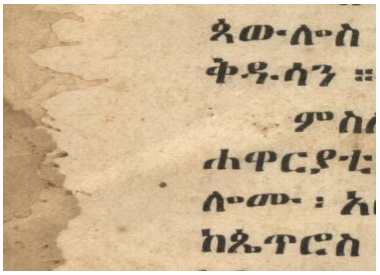


**Fig 2.2 Images showing water blotches**

**Bleed and Show-through** (see Fig. 2.3) refers to the sipping of ink from one side of a page to the other. Observation shows that it can be quite damaging, showing intensity levels that can be even darker than the true valid historical symbols (characters) in the foreground. Fig. 2.3 presents images containing exclusively bleed-through, to demonstrate how damaging it can be [39].

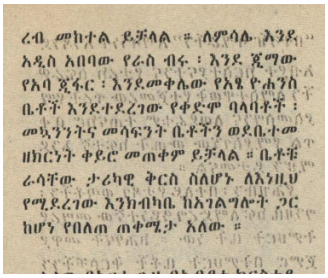


**Fig 2.3 Images showing bleed-through**

### 2.5.2 Foreground Degradation

Degradation on the foreground generally leads to broken or touching foreground objects.Age effects can affect the ink components of a document. Many chemical effects can occur leading to ink disappearance and some gaps can even appear in the document image causing significant loss of data and therefore affecting the document's content. For instance, gaps create regions with homogeneous colors. They incur a complete loss of data, whereas semitransparent blotches can preserve part of the original data. In the case of text document treatment, filling in the gap is rather complicated to proceed. Sometimes, historians must contribute. Foreground degradations are also sometimes introduced when attempting to correct symbols (characters), either by covering them with white paint or by scratching them [39]. Degradations of this kind are shown in Fig. 2.4

**Fig 2.4 Foreground (content) degradation**

### 2.5.3 Global Degradation

This type of degradation affects documents in their entirety. It refers to a transformation that can be observed in a document as a whole, i.e., without affecting uniquely the foreground or the background. This transform can act either on the localization of the pixel (skew, degraded curve) or on its value (transformation of the color). Geometrical degradation is a type of global degradation very common when scanning thick documents. The paper surface of these documents in the course of scanning is usually curved. The presence of this curvature leads to warped words appearing around the book spine area and to a non-uniform illumination. Time effect is another degradation included in this class [39][40]. Fig. 2.5 presents samples of global degradation in ancient historical documents.

**Fig 2.5 Document image showing global degradation**

## 2.6 Importance of document image restoration

With the advent of modern electronic gadgets the scope of document imaging has increased. Document image analysis systems are becoming increasingly visible in everyday life. For instance, one may be interested in processing, storing, understanding a class of document images [36].

Restoration of historical document images has many applications in enhancing the performance of character recognizers as well as in book readers used in digital libraries. Often, along with the restoration, one also looks for enhancement of the resolution. Text observed from these sources is often low-resolution degraded images, and requires restoration and resolution expansion in order to improve the performance of document image recognition and retrieval. Moreover, these imperfect images may be inadequate for subsequent human use. The visual and recognition ability fall due to these effects. The accuracy of today's document analysis algorithms falls abruptly when image quality degrades even slightly [35]. Significant improvement in accuracy on hard problems now depends as much, or more, on the size and quality of training sets as on algorithms and hardware [35].

Restoration is needed not only to improve the appearance of a document but also to improve the results of further segmentation and document image understanding operations. By clearing artifacts from the images there is less room, in the future, for misinterpretation.

## 2.7 Document image restoring Approaches

The area of document image restoration deals with recovering image information that has been degraded. In other words ***document image restoration*** means the removal or reduction of degradations that were incurred when the image was obtained. This degradation can be blurring due to the optical systems, image motion, noise due to the electronic and photometric sources, and unwanted information in the image such as show-through. To design a document image restoration system it is necessary to quantitatively characterize the image degradation effects of the physical imaging system, the image digitizer, and the image display and then undo the degradation to obtain the restored image [80].

There are two basic approaches to the modeling of image degradation effects [80]: ***a priori*** modeling and a ***posteriori*** modeling. The difference in the two modeling methods is in the manner in which information about the degradation is gathered. For an ***a priori*** modeling the response of imaging system, digitizer and display are measured on an arbitrary image field and

for a ***posteriori*** modeling the model is developed for the image degradation based on the image to be restored.

As noted by Pathak [80], the restoration techniques can be divided into three techniques, such as Spatial image restoration techniques, point and spectral restoration techniques, and geometric transformation

Pathak [80] pointed out that spatial and geometric transformations are the most widely used image restoration techniques. Point and spectral techniques are not used most of the cases because it is a post-processing correction of the sensor signal and pre-processing correction of the display signal. As a result of that, the source degradation separated into the spatial and point effects.

### 2.7.1 Spatial Image Restoration Techniques

This the most common technique used for image restoration [80]. It is used to compensate for image blur and additive noise effect caused by the sensor amplifier. The additive noise is independent of the image field. It is assumed that image blurring is modeled as a superposition of the operation with impulse response h(x,y) (*see Figure 2.6*).

The model of a degraded image can be represented as follows [80]:

$$f'(x,y) = f(x,y) * h(x,y) + n(x,y) \quad \ldots\ldots\ldots (2.1)$$

Where f' is the degraded image, f is the original image or the ideal image, h is the point spread function and n is an additive noise.

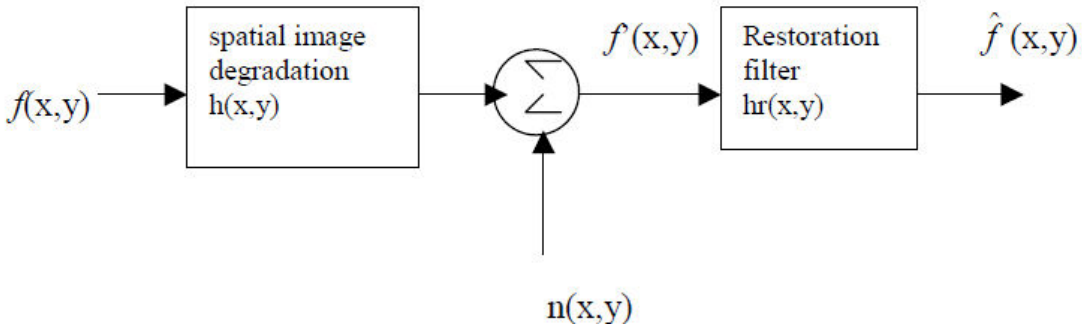



**Fig 2.6 An image degradation and restoration model**

The objective of the restoration is to produce an array of samples $\hat{f}$ (x,y) that are estimates of points on an ideal input image field $f$ (x,y) obtained by a perfect digitizer. To produce a digital restoration model it is necessary to relate the physical image sample to the ideal image points. There are various methods of restoration, including Inverse filter, Wiener convolution filter, Wiener de-convolution filter, recursive filter, least mean square filter etc [80].

### 2.7.2 Geometric Transformation

Geometric transformations are different from the above discussed restoration techniques in the sense that it modifies the spatial relationship between pixels in an image by stretching the sheet according to some predefined set of rules. There are two types of Geometric transformations. One is a **spatial transformation**, which defines the rearrangement of the pixel on the image plane by spatially relocation of pixel tie point, and the other is a **gray level interpolation**, which deals with assigning a gray level to pixel based on the nearest integer approach.

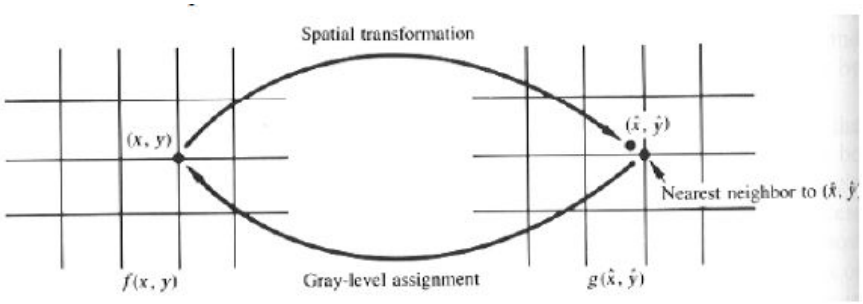



**Fig 2.7 Geometric transformation**

In general, the spatial restoration techniques are done by manipulating on a given specific pixel on the document images where as geometric transformation work by transform the whole document images to a nearest location.

## 2.8 Amharic language

In Ethiopia, more than 80 languages are used in day-to-day communication. Among them Amharic is the dominant one in that it is spoken as a mother tongue by a number of the population and it is the most commonly learned second language throughout the country. It is also the official language of the country and medium of communication and working language in most of the regional states [41].

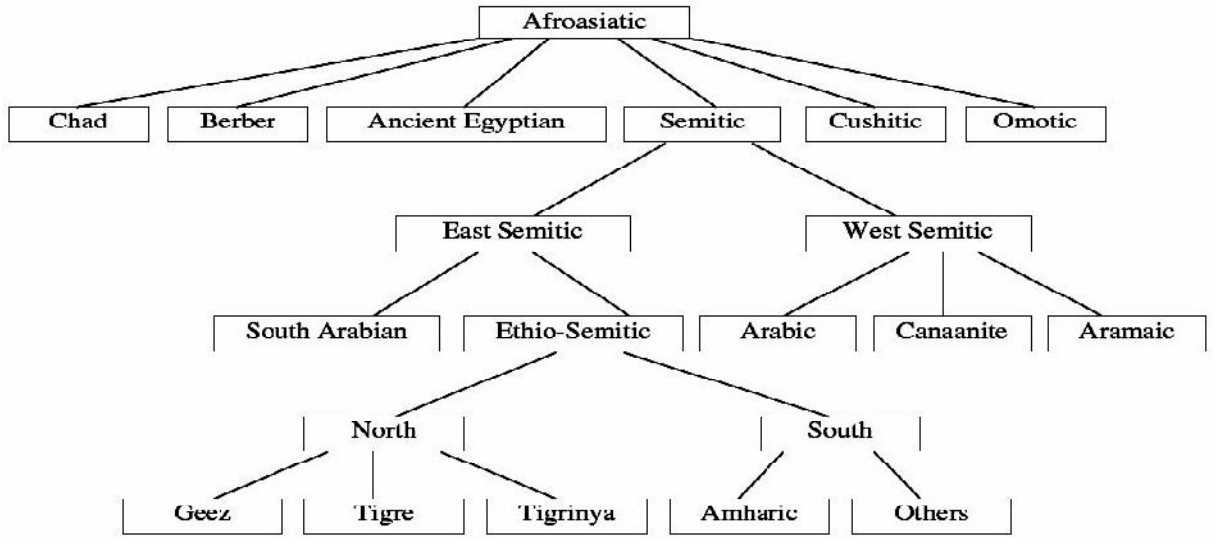



**Fig 2.8 The Genetic structure of Amharic language**

The Ethiopic script originated from the Ge'ez alphabet around 300 A.D. and is used for writing in the various languages in Ethiopia and Eritrea, including Amharic, Tigre and Tigrigna. Ethiopic script is designed as a meaningful and graphic representation of knowledge. Ethiopic script is a component of the African knowledge systems and one of the signal contributions made by Africans to the world history and cultures. It is created to holistically symbolize and locate the cultural and historical parameters of the Ethiopian people [46].

As you can see from Fig 2.8 the Ethiopian languages are divided into four major language groups, such as Cushitic, Omotic, Nilo-Saharan and Semitic. Semitic languages are spoken in northern, central and eastern Ethiopia (mainly in Tigray, Amhara, Harrari, and northern part of SNNP). The Cushitic languages are mostly spoken in central, southern and eastern Ethiopia (mainly in Afar, Oromia and Somali regions). The Omotic languages are predominantly spoken between the lakes of Southern Rift Valley and the Omo River. The Nilo-Saharan languages are largely spoken in western parts of the country along the border with Sudan (mainly in Gambella and Benishangul - Gumuz regions) [46]. Amharic belongs to the Semitic family of languages [6] [34].

### 2.8.1 Amharic writing system

Most languages spoken in the world have their own writing system and every sound is represented by some set of strokes. These combinations of strokes make up the character set of the language. The character sets or symbols used to represent the sounds of a language are called scripts of that language. After formulating and being modified through the set of characters through time, the speakers of the language will get used to these symbols so that they can use the written language approach to communicate as well as to preserve knowledge [43].

The Amharic writing system is considered as a syllabic system rather than alphabetic and uses Ethiopic script for writing. In the Amharic syllabic writing system, each character stands for a syllable rather than a single sound that means it allows anyone to write Amharic texts if s/he can speak Amharic and has knowledge of the Amharic alphabet [45].

Like many Semitic languages, Amharic uses triconsonantal roots in its verb morphology. The result of this is that a fluent speaker of Amharic can often decipher written text by observing the consonants, with the vowel variants being supplemental detail [46]. The other important feature of Amharic writing system is that it is written from left to right and there is no distinction between upper and lower case letters.

Amharic writing system consists of thirty four characters (called "fidel"/ "ፊደል") as a core characters. The thirty four core characters occur in seven orders, each of which represent syllable combinations consisting of a consonant and following vowel (see Fig 2.9). In addition to this, in the Amharic syllabic writing system, each character stands for a syllable rather than a single sound. Thus, the Amharic writing system is often called a syllabic rather than an alphabetic which allows anyone to write Amharic texts if they can speak Amharic and has knowledge of the Amharic alphabet. The non-basic forms are derived from the basic forms by more-or-less regular modifications. Part of an alphabet with their seven orders is shown in Figure 2.9 and the full Amharic script (FIDEL) set is annexed in the appendix V.

| | Base Sound | Orders | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 1st (ä) | 2nd (u) | 3rd (i) | 4th (a) | 5th (e) | 6th (ə) | 7th (o) |
| 1 | h | ሀ | ሁ | ሂ | ሃ | ሄ | ህ | ሆ |
| 2 | l | ለ | ሉ | ሊ | ላ | ሌ | ል | ሎ |
| 3 | h | ሐ | ሑ | ሒ | ሓ | ሔ | ሕ | ሖ |
| 4 | m | መ | ሙ | ሚ | ማ | ሜ | ም | ሞ |
| 5 | s | ሠ | ሡ | ሢ | ሣ | ሤ | ሥ | ሦ |
| . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| 32 | f | ፈ | ፉ | ፊ | ፋ | ፌ | ፍ | ፎ |
| 33 | p | ፐ | ፑ | ፒ | ፓ | ፔ | ፕ | ፖ |
| 34 | v | ቨ | ቩ | ቪ | ቫ | ቬ | ቭ | ቮ |

**Figure 2.9 Sample core character and their seven orders**

### 2.8.2 Amharic numeration system

Amharic numeration system consists of basic single characters for one to ten, for multiple of ten (twenty to ninety), hundreds and thousands. These numerals are derived from the Greek numerals with some modifications; each has a horizontal stroke below and above. In the system,

there is no symbol for representing zero value and it is not a place value system, thus arithmetic computation using this system is very difficult. Consequently, in most printed document Hindu-Arabic numerals are used [42]. Table 2.1 shows Amharic numerals.

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

**Table 2.1 Amharic versus Arabic numeric system**

### 2.8.3 Amharic punctuation marks

Amharic punctuation marks consist of different symbolic representations. Mostly used punctuation marks are: the basic word divider, ሁለት ነጥብ-hulet netib, which has two dots arranged like a colon (፡) and a sentence ending is represented using, ዐራት ነጥብ- arat netib, four square dots arranged in a square pattern (።). Some others equivalent to comma represented as, ነጠላ ሰረዝ--netela serez, (፣) and semicolon represented, ድርብ ሰረዝ-derib serez,(፤). Question mark, quotation mark and exclamation mark are represented as, ጥያቄ ምልክት-tiyake milikit, (?), ትምህርተ ጥቅስ-- timihrite tiks, (« ») and ትምህርተ አንክሮ-- timihrite ankro (!), respectively [42].

### 2.8.4 Labialized Characters

In general, Amharic writing system has core characters, labialization, numerals and punctuation marks which brings the total numbers of character in the script to 349 [6] [34] [44] [46] [45] [47]. Table 2.2 shows summary of the number of characters in each group.

| Type of Amharic Characters | Total Characters |
|---|---|
| Basic characters | 231 |
| Labialized characters | 89 |
| Punctuation marks | 9 |
| Numbers | 20 |
| **Total** | **349** |

**Table 2.2 Total Number of Characters in Amharic Alphabet ፊደል (Fiedel)**

## 2.9 Challenges in Amharic writing system

In Amharic language, a number of problems are observed with the writing systems that have an effect to produce words with structural variation, though they have the same meaning and sound. The major problems are summarized below:

- ***Redundancy of Consonants in Different Forms***

In Amharic Alphabets there are some different symbols having the same pronunciation (sound). The presence of these redundant characters with the same sound in the language creates problem, especially in terms of matching words during the searching process. Literally different words can be formed by combining the different forms of the same sound characters. For example, the word "sun" can be written as "ፀሀይ", "ፀሃይ", "ፀሐይ", "ፀኀይ", "ጸሀይ", "ጸሃይ", "ጸሐይ", "ጸኀይ", etc. all mean the same, although they are written differently.

- ***Multiple Writing Systems for Single Amharic Word***

The rule to Amharic language users generally follow in their writing system is that if the alphabet in a word sounds right when read aloud, then it is written. There are different ways of writing a single word due to various reasons such as regional dialects and various ways of writing loan words. Regional dialects have their own impact in word formation in the basic level where the words are more likely to be written by following their spoken form. For example, "ዓዜ" vs. "ዓጤ", "እነዚህ" vs. "እነኚህ", etc. The absence of restricted rules leads to a number of problems to develop efficient Amharic information retrieval system.

- ***Compound Words***

The Amharic writing system uses multitudes of ways to denote compound words and there is no agreed upon spelling standard for compounds. These words can be written as two separate words or as a single word using 'hulet netib' (:) in between. For instance, the word "library" can be written as "ቤተመጻሕፍት" or "ቤተ፡መጻሕፍት" or "ቤተ መጻሕፍት".

- ***Abbreviations***

In Amharic writing system there is no consistency of spelling abbreviations. For example, when abbreviating the phrase **ዓመተ ምህረት** (in the year AD), one can find **ዓ.ም**, **ዐ.ም** or **አ. ም**; **አዲስ አበባ** (Addis Ababa the capital city of Ethiopia) is also written as **አአ**, **አ.አ** abbreviations. The use of dot is not consistent throughout the writing system [23].

## 2.10 Related works

Due to limitations of optical character recognition systems a number of attempts are made by researchers to avoid the use of character recognition for various document image retrieval applications. Consequently, the research direction towards retrieving information without recognition from document images considers printed or handwritten documents which are encoded in different languages. In this thesis, we attempt to review researches done internationally and locally on document image retrieval.

### 2.10.1 International research work

Banerjee, Namboodiri, and Jawahar [2] propose an approach to restore severely degraded document images using a probabilistic context model. Un-like traditional approaches that use previously learned prior models to restore an image; they are able to learn the text model from the degraded document itself, making the approach independent of script, font, style, etc. They model the contextual relationship using a Markov Random Field (MRF). The ability to work with larger patch sizes allows them to deal with severe degradations including cuts, blobs, merges and vandalized documents. Their approach can also integrate document restoration and super-resolution into a single framework, thus directly generating high quality images from degraded documents.

Document images are collected from various sources including magazines and books, and comprehensively demonstrate the robustness and adaptability of the approach. It works well with document collections such as books, even with severe degradations, and hence is ideally suited for repositories such as digital libraries. The restoration also improves the recognition results of any off-the-shelf OCR system. To verify this, they ran the Tesseract-2.01 OCR from Google on degraded English book containing 40 pages with close to 50, 000 words and 237, 000 characters which resulted in an error rate of 3.7%.However, after restoration by their proposed algorithm, the error rate further reduced to 1.9%. They also noted that the current approach primarily uses a content model that is learned from the input document. Integration of the approach with a complementary mechanism that models the nature of degradations could further improve the restoration performance. Another potential direction is to combine recognition with restoration in an iterative fashion.

Gangamma and Murthy [48] proposed a method combining two powerful image processing techniques, spatial filtering technique and gray scale mathematical morphology. The proposed method takes degraded historical document as input and performs series of steps such as grabbed image is converted into gray scale image and then binarized the degraded document image and apply bilateral filter to filter the image. Mathematical morphology which is based on set theory approach uses simple operations which are computationally less complex is applied in bridging the gap between broken parts of the character. The proposed method is used to enhance the palm

script and epigraphically script images. Experimentation has been performed on the set of more than 200 images of various sizes. The camera grabbed images are used with size varying from 3000x4500 to 300x450. Length and width of the palm scripts are varying from 40 inch to 2 inch and from 4-5 inch to 2 inch respectively. The proposed method eliminates noise, uneven background and enhances the contrast of the script image and the Performance of the proposed method proved to be better than mean and Gaussian methods in-terms of clear uniform background and foreground with enhanced character appearance. The also noted that the enhanced document image can be used further to segment the document into lines, words and character for recognition purpose.

Tonazzini, Bedini, and Salerno[49] propose a novel approach to restoring digital document images, with the aim of improving text legibility and OCR performance. These are often compromised by the presence of artifacts in the background, derived from many kinds of degradations, such as spots, underwritings, and show-through or bleed-through effects. So far, background removal techniques have been based on local, adaptive filters and morphological–structural operators to cope with frequent low-contrast situations. For the specific problem of bleed-through/show-through, most work has been based on the comparison between the front and back pages. This, however, requires a preliminary registration of the two images.

They presented preliminary results of the application of independent component analysis (ICA) techniques to the problem of the separation of overlapped texts in documents showing bleed-through or show-through and in palimpsests. Their approach is the first result of a study on the possibility of formulating the problem as a particular kind of blind source separation, which is a well established discipline in signal processing, but still at the initial stage in the field of image processing and especially in the area of document analysis. It involves a linear, noiseless data model where each color channel of the input image is a mixture of all the patterns to be extracted. On the one hand, for the extraction of hidden text, the advantages of exploiting multiple channels over the analysis of single-channel images have been confirmed. Within the approaches they operate some linear combination of the input channels; their method has additional advantage that the related coefficients do not need to be known in advance. On the other hand, the possibility of rejecting patterns interfering with the main text will certainly be useful for improving legibility. This means, it both legible by a human reader and by an OCR system. The

result show that independent component analysis (ICA) techniques do not work well for documents showing bleed-through or show-through and in palimpsests. As a result of this, they recommend a better technique that is suitable for such type of document image.

### 2.10.2 Local research work

A novel research on document image retrieval without explicit recognition for Amharic language in addition to English and Indian languages was conducted by million [6]. The motivations were the lack of robust OCR system for Amharic language and designing OCR is also a long term process for retrieving information from document images. The study proposed, on the one hand, an effective word image matching called dynamic time warping (DTW), and on the other hand, optimal combined features of transition profile, lower profile with order moment and upper profile scheme that achieves high performance in presence of script variability, printing variations, degradations and word-form variations. Datasets containing a total of more than 800, 000 word images in English, Hindi and Amharic languages were prepared to conduct extensive experiments. The dataset contains basic words with their morphological variants generated using degradation models such as salt and pepper, cuts, blobs and erosion of pixels which are printed using various fonts, styles and sizes.

Test result on degraded text images shows that performance varies depending on the degradation type. On average, 92.52%, 95.49%, 89.51% and 93.61% F-scores are obtained on cuts, salt and pepper, blobs and erosion, respectively. The average performance of the proposed scheme on the various fonts, styles and sizes is 91.62% F-score. The researcher clearly stated the need to apply advanced image pre-processing techniques (image restoration and skew correction) for document analysis since printed document images are more of historical and poor in quality and document image processing algorithms for document image collections are missing.

Mesfin [7] also attempt to design a retrieval system that can search for relevant document images from scanned Amharic document image corpus by accepting a query from the user depending on image features only. Amharic document images were scanned in grayscale with 300 dpi intensity and fixed threshold method was used. Two phased segmentation algorithm i.e. line segmentation

followed by word segmentation was implemented. Line and words in a document are identified using horizontal and vertical boundary segmentation.

Experiment is carried out on 121 scanned Amharic documents containing 483 pages and 109,238 words that are selected from printed legal documents and news items, among which 28 word queries were selected for testing. All documents in the dataset have a font size of 12, a font type of Power geez Unicode 1 and a plain style. One of the limitations of this research is, it does not consider ancient and handwritten documents. The retrieval effectiveness of the system was measured using precision, recall and F-measure. Based on performance analysis, the highest average F-measure of 57.08% was achieved using parallel bar feature extraction method and Euclidean similarity measure. In order to increase the performance of the proposed retrieval system, pre-processing techniques like image restoration, font resizing and font conversion are recommended.

Based on Mesfin's [7] recommendation, Abreham [8] works aim to design an efficient indexing scheme for organizing Amharic image corpus and enhance searching for relevant documents. To extract features, word shape analysis of vertical bar pattern is performed. The extracted feature uniquely identifies each of the word in the document collection. To measure the similarity between images, Cosine similarity measurement was used which was also used to detect suffix and prefix of word images. The suffix and prefix detection at the time of comparing two image terms shows 85.3% average accuracy. Inverse document frequency (IDF) was computed to remove common word images or stop words which occurred in more than 80% of the documents.

The efficiency and effectiveness of the system is tested using test cases. The efficiency is checked by measuring the time taken to search relevant documents from the index file. Further effectiveness of the system is measured using recall, precision and F-measure. According to the result, effectiveness of the system shows F-measure value of 41.59%. The efficiency, which was measured using the average search time required before and after indexing, showed improvement by more than 26.6%.

The performance of the system is greatly affected by foreground degradation and noise that exist within real-life document images. Therefore, pre-processing techniques like image restoration,

noise removal, skew correction and normalization are recommended to be integrated to increase the effectiveness and efficiency of the system.

Continuing Abreham's [8] work, Adane [9] endeavored to design and integrate effective and efficient feature extraction and matching schemes which are insensitive to the artifacts in real life word images, such as foreground degradation, noise, word variations and difference due to font sizes, styles and types in order to enhance the performance of Amharic document image retrieval system.

Primarily, computer printed Amharic documents with various font styles, sizes and types are collected from books, magazines, and newspapers which are then converted to gray scale digital images using a flatbed scanner at 300 DPI. The document images are then binarized into digital and manageable representations. This is followed by line and word segmentation respectively. Then, features are extracted at word level using combination of three feature extraction techniques Vertical distance, Vertical projection and Lower word profile.

Based on DTW matching algorithm, the best performance of 95.52% is registered. DTW achieved good matching performance on low and medium level noisy and degraded document images. The selected matching, feature extraction and stemming techniques are integrated to the previous Amharic document image retrieval system and tested on noisy and degraded real life document images. Accordingly, the maximum F-measure of 80.46% is achieved on low level noisy (degraded) document images. On the other hand, 66.66% and 55.82% F-measures are registered on medium and high level noisy (degraded) document images, respectively.

Analysis of the performance of the system shows that it is significantly affected by the existence of noise and different levels of degradations in real-life document images. Hence, integrating image restoration and noise reduction technique improves the performance of the document image retrieval system.

Biniam [10] attempted to improve the effectiveness of the Amahric document image retrieval system by integrating noise removal scheme. Three noise reduction (or filtering) techniques; median, wiener and adaptive median filters are evaluated in combination with Otsu, Niblack and

Sauvola thresholding algorithms. A series of experiments were conducted to select the optimal combination of noise removal and thresholding techniques on low, medium, high and very high noisy document images. The performance of the system is evaluated before and after the integration of the selected pre-processing techniques in which an average overall performance of 82.37% F-measure is registered in documents having low, medium, high and very high levels of noise. However, he boldly indicates that in order to come up with a practical Amharic document image retrieval system, image pre-processing tasks such as image restoration and edge detection are vital.

Recently Kibrom[11] and Gedion[12] worked on segmentation of document images having different outputs. Kibrom[11] implement three segmentation algorithms (Normalized cut, thresholding and projection based segmentation) and the performance of segmentation algorithms are tested on historical Amharic document images of different noise levels. Test result shows that on the average Normalized cut segmentation technique outperforms thresholding and projection based segmentation algorithms.

The resulting segmentation algorithm has also been integrated to the previous Amharic document image retrieval system. Performance of the segmentation algorithm has been measured after pre-processing the noisy historical Amharic documents. To test the performance of the ADIRS the researcher selected n-query word images and their corresponding recall, precision and F-measure. It has been noted that the segmentation algorithm improves the performance of the retrieval system an average of 12.24% F-measure.

Gedion [12] further implement page segmentation algorithms namely: Hough transforms, Connected Components (CC), Horizontal Run Length Smoothing (HRLS), Dilation and Watershed. The performance evaluation showed that the integration of CC and Dilation is the best combination. Average Match Score of 0.865 in different level noisy document images, 0.93 in typewritten documents, 0.97 in documents containing pictures, 0.97 in documents containing tables and 0.45 in handwritten documents (‘kum tsehfet‘) is scored. On the average, an increase of 2.34% F-Measure is scored in different level noisy document images.

Since, historical document images are usually distorted, Kibrom and Gedion recommended the need to adopt image restoration technique for image enhancement such that the performance of ADIRS is further improved.

Pre-processing is one of the main tasks that prepare a document image for efficient and enhanced retrieval by restoring its degraded effect on it. Previously developed Amharic document image retrieval systems (ADIRS) [7] [8] [9] [10] [11] [12] lack document image restoring mechanisms. So, there is a need to identify suitable algorithm which can be applied to any type of Amharic degraded historical document images to revert back the degradation process by restoring historical Amharic document. In addition to searching with n word queries, image viewer and enhancing the interface and integrate it with ADIRS.

# CHAPTER THREE

# METHODS AND ALGORTHMS

Pre-processing is one of the major tasks that prepare document images for efficient and enhanced content analysis. In this study an attempt is made to apply image restoration techniques, in order to restore document images from degraded historical document images many phases are involved. Some of them are finding out degradation causes, establishing degradation model and reverse evolution image restoration [39].

The goal of an image restoration algorithm is to generate an estimate of the original picture prior to the degradation. The term image restoration is usually associated with minimizing or even removing image artifacts due to blurring and noise. Restoration methods are usually based on explicitly models and evaluated quantitatively [40].

In this chapter, some restoration techniques to restore degraded document images, modifying vector space model to enable accept N-words query from the user have been explored and integrated to increase the performance of the Amharic document images.

## 3.1 Architecture of the Amharic Document Image Retrieval System

The proposed architecture for the Amharic document image retrieval system is shown in Figure 3.1. Two main processes are performed: offline and online. The offline process is the indexing process that starts by scanning historical documents in order to come up with a digitized document images. Then, this document images are pre-processed to biniaze, restore and clean them. After pre-processing, the document images are segmented to identify bag of words and their feature values are extracted. This feature values are finally stored using inverted index file indexing structure.

The process of query segmentation is the same as that of image segmentation as long as the query is submitted in image form. However, the system is designed to accept the query in text form. Hence, it is necessary to convert the submitted query text into an image by rendering.

If N words query is given, the text query is converted into word level and then each word is rendered to image. Features are extracted from the processed query image and similarity measure is calculated to search for and retrieve relevant documents that match or contain the query words. Finally, the ranked lists of retrieved relevant documents are displayed back to the user. The focus of this research is represented in figure 3.1 by double rectangle and bold

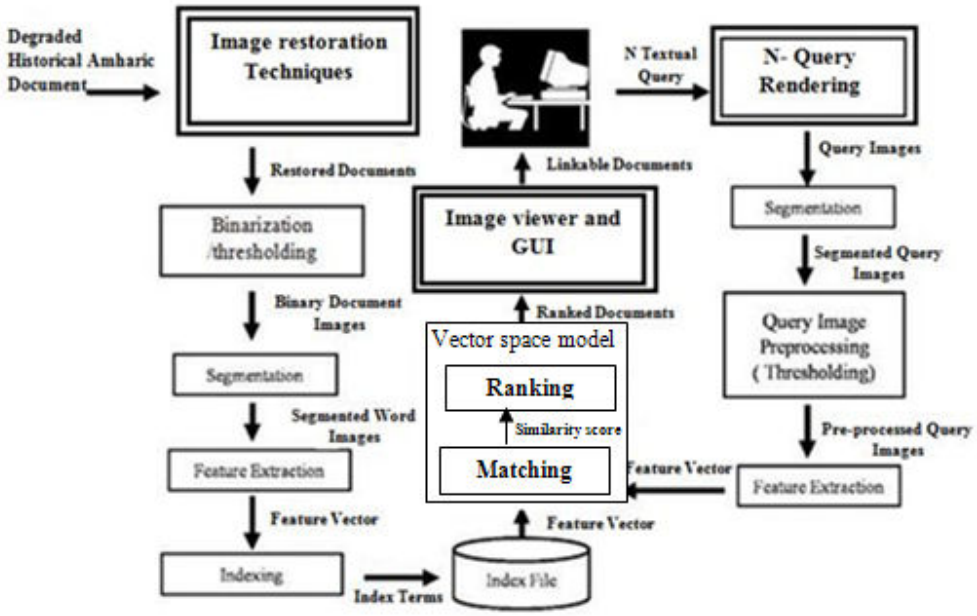



**Fig 3.1 Architecture for the Amharic document image retrieval system (ADIRS)**

As this study is a continuation of the previous works by Mesfin [7], Abreham[8], Adane [9] , Binaim [10] , Kibrom[11] and Gedion [12] ,it attempt to integrate different modules focusing on restoring the degraded historical document image in addition to searching with N words query, image viewer and enhancing the interface of ADIRS.

## 3.2 Document Image Restoring Techniques

In this study, six image restoring techniques (three from mathematical morphology and three from wavelet transformation) are explored to see their effect on historical Amharic document images.

### 3.2.1 Mathematical Morphology

Morphology is a technique of image processing based on shape and form of objects [62]. Morphological methods apply a structuring element to an input image, creating an output image at the same size. The value of each pixel in the input image is based on a comparison of the corresponding pixel in the input image with its neighbours. By choosing the size and shape of the neighbour, we can construct a morphological operation that is sensitive to specific shapes in the input image. The morphological operations can first be defined on grayscale images where the source image is planar (single-channel).

There are two basic morphological Operations: *erosion* and *dilation* [53] .Using the basic operations we can perform *opening* and *closing*. More advanced morphological operation can then be implemented using combinations of all of these to perform morphological image analysis. Morphological operations apply structuring elements to an input image, creating an output image of the same size. Irrespective of the size of the structuring element, the origin is located at its centre. The algorithms for the four mentioned operations are dependent on the **structured element (SE)** and are discussed below.

Dilation operation enlarges a region, while erosion makes it smaller. Erosion operation is a morphological operation for reducing the foreground area. The effect of this operation is shrunk

foreground. The foreground is reduced from its outer edge to its inside area. If there is a hole inside the foreground area, the hole enlarges. It uses a structuring element and it is done with a convolution operation between the image and the structuring element [53].

**Dilation**

Dilation is a transformation that produces an image that is the same shape as the original, but is a different size. Dilation stretches or shrinks the original figure. Dilation increases the valleys and enlarges the width of maximum regions, so it can remove negative impulsive but do little on positives ones.

The dilation of $A$ by the structuring element $B$ is defined by [62]:

$$A \oplus B = \bigcup_{b \in B} A_b \qquad \text{- - - - - - - - - - - (3.1)}$$

Dilation of image $f$ by structuring element $s$ is given by $A \oplus B$. The structuring element s is positioned with its origin at $(x, y)$ and the new pixel value is determined using the rule:

$$g(x,y) = \begin{cases} 1 & \text{if } s \text{ hits } f \\ 0 & \text{otherwise} \end{cases} \qquad \text{- - - - - - - - - - - (3.2)}$$

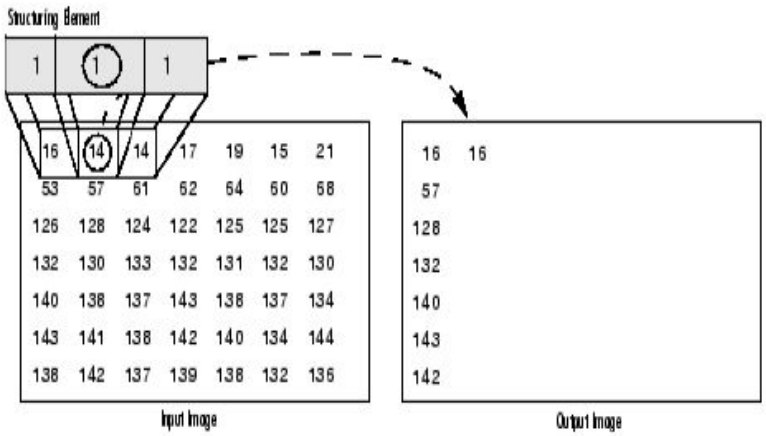



**Fig 3.2 Morphological dilation of grayscale image**

Figure 3.2 above illustrate the morphological dilation of a gray scale image. Note how the structuring element defines the neighbourhood of the pixel of interest, which is circled. The dilation function applies the appropriate rule to the pixels in the neighbourhood and assigns a value to the corresponding pixel in the output image. As shown in the figure 3.2 above, the morphological dilation function sets the value of the output pixel to 16 of the first three selected pixel values because it is the maximum value of all the pixels in the input pixel's neighbourhood defined by the structuring element. As a result, the neighbourhood pixel values change from 14 to 15 and Algorithm 3.1 below shows how dilation increases the valleys and enlarges the width of maximum regions by using a structural element (SE) who defines it.

**Algorithm 3.1 Mathematical Morphology Dilation**

**Determine the SE, including its definition domain and the value of each element.**
**Suppose m ≤ s≤ n;**
**For (each sample of the signal f(x))**
  **For (m ≤ s≤ n)**
    **Calculate $\omega$(s-m+1) =f(x+s)+g(s);**
  **End**
**Return the maximum element of $\omega$ and f $\oplus$ g(x)=max{ $\omega$};**

**Erosion**

It is used to reduce objects in the image and known that erosion reduces the peaks and enlarges the widths of minimum regions. That means, the value of the output pixel is the *minimum* value of all the pixels in the input pixel's neighbourhood.

The erosion of the binary image *A* by the structuring element *B* is defined by [62]:

$$A\theta B = \{z \in E | B_{\approx} \subseteq A\} \text{ - - - - - - - - - - - - - - } (3.3)$$

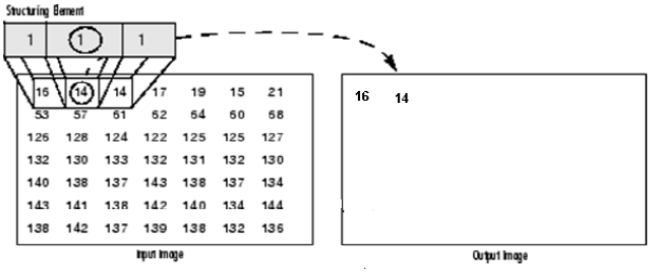



**Fig 3.3 Morphological Erosion of gray scale image**

Figure 3.3 illustrates the morphological erosion of a gray scale image. Note how the structuring element defines the neighbourhood of the pixel of interest, which is circled. The erosion function applies the appropriate rule to the pixels in the neighbourhood and assigns a value to the corresponding pixel in the output image. In the figure, the morphological erosion function sets the value of the output pixel to 14 because it is the minimum value of the three selected pixels value in the input pixel's neighbourhood defined by the structuring element. Algorithm 3.2 shows how erosion operation applied for a given image. The effect of this operation shrunk foreground and to do that it uses a structuring element and it is done with a convolution operation between the image and the structuring element.

**Algorithm 3.2 Mathematical Morphology Erosion**

**Determine the SE, including its definition domain and the value of each element.**
**Suppose m s n;**
**For (each sample of the signal f(x))**
**For (m s n)**
**Calculate (s-m+1) =f(x+s)-g(s);**
**End**
**Return the minmum element of and f g(x) = min{ };**
**End**

## Opening

*Opening* generally smoothes the contour object, breaks narrow isthmuses, and eliminates thin protrusions. Opening decreases sizes of the small bright detail, with no appreciable effect on the darker gray levels, while the closing decreases sizes of the small dark details, with relatively little effect on bright features.

The opening of *A* by *B* is obtained by the erosion of *A* by *B*, followed by dilation of the resulting image by *B* [62]:

$$A \circ B = (A \; ! \; B) \oplus B \quad \text{- - - - - - - - - - - - (3.4)}$$

**Algorithm 3.3 Mathematical Morphology Opening**

**Determine the SE, including its definition domain and the value of each element.**
**Suppose m s n;**
**For (each sample of the signal f(x))**
**For (m s n)**
**Calculate (s-m+1) =f(x+s) - g(s);**
**End**
**Return the minimum element of and ε (x)=min{ };**
**End**

**For (each sample of the signal ε (x))**

**For (m s n)**
**Calculate (s-m+1) = ε (x+s) + g(s);**
**End**
**Return the maximum element of and (x) = max{ };**
**End**

Algorithm 3.3 shows how mathematical opening is performed on a given image. As we can see from the algorithm first it processes the erosion part followed by dilation.

**Closing**

Closing of an image is the reverse of opening operation. *Closing* also tends to smooth sections of contours but, ass opposed to opening; it generally fuses narrow breaks and long thin gulfs, eliminates small holes, and fills gaps in the contour.

The closing of *A* by *B* is obtained by the dilation of *A* by *B*, followed by erosion of the resulting image by *B* [62]:

$$A \bullet B = (A \oplus B) \; ! \; B \quad \text{- - - - - - - - - - - - (3.5)}$$

**Algorithm 3.4 Mathematical Morphology Closing**

**Determine the SE, including its definition domain and the value of each element.**
**Suppose m ≤ s≤ n;**
**For (each sample of the signal δ(x))**
**For (m ≤ s≤ n)**
**Calculate ϖ(s-m+1) =δ (x+s) - g(s);**
**End**
**Return the minimum element of ϖ and (x)=min{ϖ};**
**End**

**For (each sample of the signal *f*(x))**

**For (m ≤ s≤ n)**
**Calculate *ω*(s-m+1) = f (x+s) + g(s);**
**End**
**Return the maximum element of *ω* and *δ*(x) = max {*ω*};**
**End**

Algorithm 3.4 above shows how mathematical closing is performed on a given image. As we can see from the algorithm first it processes the dilation part followed by erosion.

### 3.2.2 Wavelet Transformation

The fundamental idea behind wavelets is to analyze according to scale. Indeed, some researchers in the wavelet field feel that, by using wavelets, one is adopting a whole new mindset or perspective in processing data. Wavelet algorithms process data at different scales or resolutions [57].

Wavelet analysis can be used to divide the information of an image into approximation and detail sub-signals. The approximation sub-signal shows the general trend of pixel values [57].

The wavelet analysis procedure is to adopt a wavelet prototype function, called an analyzing wavelet or mother wavelet. Temporal analysis is performed with a contracted, high-frequency version of the prototype wavelet, while frequency analysis is performed with a dilated, low

frequency version of the same wavelet because the original signal or function can be represented in terms of a wavelet [57] [58].

Wavelets have far more extensive uses. They can be used to process and improve signals, in fields such as medical imaging where image degradation is not tolerated as they are of particular use. They can be used to remove noise in an image, for example if it is of very fine scales, wavelets can be used to cut out this fine scale, effectively removing the noise [58].

There are different types of wavelet families whose qualities vary according to several criteria. The main criteria is the regularity, which is useful for getting nice features, like smoothness of the reconstructed signal or image, and for the estimated function in nonlinear regression analysis [59][61]. They may also be associated with these less important properties:

- The existence of an explicit expression.
- The ease of tabulating.
- The familiarity with use.

Some of the Wavelet Families included in the MATLAB are:

'haar' : Haar wavelet.

'db' : Daubechies wavelets.

'sym' : Symlets.

'coif' : Coiflets.

'bior' : Biorthogonal wavelets

The wavelet transform plays an extremely crucial role in reconstruction. Information about signals resulting from a selected process can be based upon signal decomposition by a given set of wavelet functions into separate levels or scales resulting in the set of wavelet transform coefficients. These values can be used for image reconstruction.

The mathematical approach to the discrete wavelet transform (DWT) is based on the fact that a function f (t) can be linearly represented as [57][58] :

$$f(t) = \sum_k a_k \psi_k(t) \quad \text{----------} \quad (3.6)$$

Where $a_k$ are the analysis coefficients and $\psi_k$ the analyzing functions

**The whole method consists of the following steps:**

• Signal decomposition using a chosen wavelet function up to the selected level and evaluation of wavelet transform coefficients
• The choice of threshold limits for each decomposition level and modification of its coefficients.
• Signal reconstruction from modified wavelet transforms coefficients.

Results of this process depend upon the proper choice of wavelet functions, selection of threshold limits and their use. It is necessary to know about the two general categories of thresholding. They are hard- thresholding and soft thresholding types. In contrast to hard thresholding, soft thresholding causes no discontinuities in the resulting signal. In MATLAB, by default, soft thresholding is used for de-noising and hard thresholding for compression and image reconstruction [60].

**Algorithm 3.5: Wavelet Transformation**

- Choice of a wavelet function (e.g. Haar, symmlet, etc)
- Choice number of levels or scales for the decomposition (1 to N).
- Estimation of a threshold(hard or soft depend on the task we want to perform)
- Perform iteratively reconstruct by using wavrec () function.

### 3.2.3 Otsu thresholding

Otsu's global threshold method [10] finds the global threshold $t$ that minimizes the intra-class variance of the resulting black and white pixels. For all the pixels inside $l$, Otsu's threshold $T$ is calculated to divide the pixels into two clusters. If the two estimated cluster means $\hat{\mu}_1$ and $\hat{\mu}_2$ are further apart than a user-specified limit, $\|\hat{\mu}_1 - \hat{\mu}_2\| \geq l$, then the pixels inside $S$ are binarized using the threshold value $T$. When $\|\hat{\mu}_1 - \hat{\mu}_2\| < l$, all the pixels inside $S$ are assigned to the class with the closest updated mean value [67]. This is a standard binarization technique and was implemented using the built-in MATLAB function "graythresh" [38]. Then, the binarization is formed by setting $b_i = 1$ if $x_i >= t$ and $b_i = 0$ if $x_i < t$ [10].

The default binarization technique used in MATLAB Image Processing Toolbox is Otsu thresholding. The function *BW = im2bw(I, level)* converts the grayscale image *I* to a binary image. The output image *BW* replaces all pixels in the input image with luminance greater than level with the value 1 (white) and replaces all other pixels with the value 0 (black) [10]. The function *graythresh* can be used to compute the level which is a value between 0 and 1.

We wrote a function that calculates the level using *graythresh* function and supply the result to *im2bw* Otsu binarization method as shown in algorithm 3.6.

**Algorithum 3.6: Otsu Thresholding**

```
function ThresholdedImage = OtsuThresholding(FilteredImage)

% Apply Otsu thresholding to the filtered image using graythresh and im2bw()

level = graythresh(FilteredImage);

% Use the variable 'level' as an argument in im2bw()

ThresholdedImage = im2bw(FilteredImage, level);
```

## 3.3 Vector Space Model

The representation of a set of documents as vectors in a common vector space is known as the *vector space model* and is fundamental to a host of information retrieval operations ranging from scoring documents on a query, document classification and document clustering. The basic premise of adopting the vector space model is that the various information retrieval objects are modeled as element of vector space. Specifically terms, documents, queries, concepts, and so on are all vectors in the vector space [34][15].

The vector space model is accomplished by assigning non-binary weights to index terms in queries and in documents. These term weights are ultimately used to compute the degree of similarity between each document stored in the system and the user query. By sorting the retrieved documents in decreasing order of this degree of similarity, the vector model takes into consideration documents which match the query terms only partially. The main resultant effect is that the ranked document answer set is a lot more precise (in the sense that it better matches the user information need) than the document answer set retrieved by the Boolean model [10][12].

**Definition:**

*For the vector model, the weight* $w_{i,j}$ *associated with a pair* ($k_i$ , $d_j$) *is positive and non-binary. Further, the index terms in the query are also weighted. Let* $w_{i,q}$ *be the weight associated with the pair [ki , q], where* $w_{i,q} \geq 0$. *Then, the query vector* $\vec{q}$ *is defined as* $\vec{q} = (w_{1,q}, w_{2,q}, \ldots, w_{t,q})$ *where t is the total number of index terms in the system. As before, the vector for a document dj is represented by* $\vec{d} = (w_{1,j}, w_{2,j}, \ldots, w_{t,j})$.

Therefore, a document *dj* and a user query *q* are represented as *t-dimensional* vectors as shown in Figure 3.4. The vector space model proposes to evaluate the degree of similarity of the document *dj* with regard to the query *q* as the correlation between the vectors $\vec{d_j}$ and $\vec{q}$. This correlation can be quantified, for instance, by the cosine of the angle between these two vectors [10]. That is,

$$sim(d_j, q) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}} \quad (3.7)$$

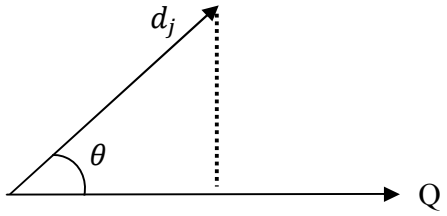



Figure 3.4: The cosine of $\theta$ is adopted as *sim(dj,q)*

where $|\vec{d}_j|$ and $|\vec{q}|$ are the norms of the document and query vectors. The factor $|\vec{q}|$ does not affect the ranking (i.e., the ordering of the documents) because it is the same for all documents. The factor $|\vec{d}_j|$ provides normalization in the space of the documents. Since $w_{i,j} \geq 0$ and $w_{i,q} \geq 0$, *sim(q, dj)* varies from 0 to +1. Thus, instead of attempting to predict whether a document is relevant or not, the vector space model ranks the documents according to their degree of similarity to the query.

In this study, vector space model is used to ease searching using multiple word queries in the image space.

## 3.4 Performance Measure

One of the main goals of this research is to integrate image restoring method to the already developed Amharic DIRS. Therefore, to quantitatively assess the strength and quality of the restored images Mean Squared Error (MSE) and Peak Signal to Noise Ratio (PSNR) are used.

### 3.4.1 Mean Square Error (MSE) and Peak Signal to Noise ratio (PSNR)

The Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR) are the two error metrics used to compare image compression quality [50]. The **Mean Square Error** is the cumulative squared error between the degraded image and the original image (see equation 3.7) [51].

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[X(i,j) - Y(i,j)]^2 \quad \text{- - - - - - - - - (3.7)}$$

Where X(i,j) is the original image, Y(i,j) is the approximated version and M,N are the dimensions of the images.

**Peak Signal to Noise Ratio (PSNR)** is the ratio between the maximum possible power of a signal and the power of corrupting that affects the fidelity of its representation [50]. Saba et.al [52] defined PSNR as the ratio of the variance of the noise-free signal to the mean-squared error between the noise-free signal and the distorted signal.

$$PSNR = 20 * \log_{10}^{(255 / \text{sqrt (MSE)})} \quad \text{- - - - - - - - - - - - - - - (3.8)}$$

PSNR is usually expressed in terms of the logarithmic decibel scale, ‘dB’ for short. MSE and PSNR values are inversely related in expressing the quality of the image the lower value for MSE means lesser error, and as seen from the inverse relation between the MSE and PSNR, this translates to a high value of PSNR. Logically, a higher value of PSNR is good because it means that the ratio of Signal to Noise is higher. Here, the 'signal' is the original image, and the 'noise' is the error in reconstruction. So, if the document registers a lower MSE and a high PSNR, you can recognise that it has a better quality [50].

### 3.4.2 Precision, Recall and F-measure

An information retrieval system returns relevant documents that satisfy the information need of users' query. In order to present a set of ranked documents, the performance in terms of effectiveness and efficiency should be measured. In this Amharic DIRS, we measure the effectiveness of the system before and after integrating image pre-processing module to the existing system. According to Manning et. al. [34], the two most frequent and basic measures for information retrieval effectiveness are precision and recall.

While Precision is the fraction of retrieved documents that are relevant, Recall is the fraction of relevant documents that is retrieved from the total number of relevant document in the collection [15]. Delalandre [52] presents a clearly understandable equation of precision and recall as follows.

$$Precision = \frac{|\{relavant\ documents\} \cap \{retrieved\ documents\}|}{|\{retrieved\ documents\}|}$$

$$Recall = \frac{|\{relavant\ documents\} \cap \{retrieved\ documents\}|}{|\{relevant\ documents\}|} \quad \text{- - - - (3.9)}$$

The advantage of having the precision and recall value is that one is more important than the other in many circumstances. Recall is a non-decreasing function of the number of documents retrieved. On the other hand, in a good system, precision usually decreases as the number of documents retrieved increase. In general, we want to get some amount of recall while tolerating only a certain percentage of false positives [52].

A single measure that trades-off precision versus recall is the F-measure, which is the weighted harmonic mean of precision and recall [9].

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \quad \text{- - - - - - - - (3.10)}$$

# CHAPTER FOUR

# EXPERIMENTATION AND DISCUSSION

This study has a great contribution toward enhancing the performance of the Amharic document image retrieval system. An experiment was undertaken to select and integrate suitable restoration technique for degraded historical Amharic document images. In addition, vector space model with N-words query searching capability is enabled, besides integrating image viewer with the interface of Amharic Document Image Retrieval System (ADIRS).

For the experimentation Dell Desktop computer with specification Intel® Core™ i3 CPU 550 @ 3.20 GHz (4 CPUs), 4GB RAM and Windows® 8 professional edition operating system was used. MATLAB™ image processing toolbox 7.0 and Java™ programming language using Net Beans IDE 7.1.2 are used for developing and integration.

## 4.1 Dataset preparation

In this research, different Amharic historical documents collection called "Qum Tsehfet" from **IES** (Institute of Ethiopian Studies), different historical letters written collection for Ras Teferi former king of Ethiopia and old written histories from National Achieve and Library Agency (**NALA**) are digitized. A camera with a resolution of 300 depth-per-inches (DPI) is used. 300 DPI is the preferred and efficient level for historical documents because it does not tend to break thin lines or fill gaps [49].

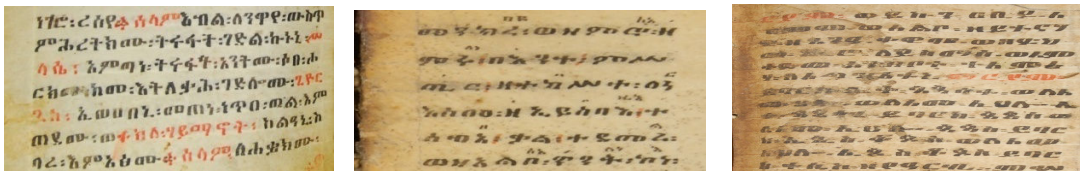
To evaluate the performance of the system, 2987 word images (1145 from low level degradation, 992 from medium level degradation and 850 from high level degradation) are collected from different historical Amharic document sources. Based on the level of degradation prevalent in the document images and the pixel intensity[53][54], we classified them into low, medium, and high level degradation by comparing every degraded image with the restored(reconstructed) image of itself and measuring the difference using MSE (Mean Square Error) as depicted in Figure 4.1 and Figure 4.2. The type and level of image and the maximum MSE is presented as follows.

- *Low level:* if mean square error (MSE) of the document image is less than 0.0462, then it is taken as low level degraded document image (Figure 4.1 (a & b)).
- *Medium level:* if mean square error (MSE) of the document image greater than 0.0465 and less than 0.0760, then it is taken as medium level degraded document image (Figure 4.1 (c & d)).
- *High level:* if mean square error (MSE) of the document image greater than 0.0770, then it is taken as high level degraded document image (Figure 4.1 (e & f)).

(a) ( c) (e)

**Fig 4.1 Different levels of degraded Amharic historical letter**

(a) Low level degraded historical letter,(c) Medium level degraded historical letter ,(e) High level degraded historical letter



(b) (d) (f)

**Fig 4.2 Different levels of degraded Qum Tsehfet**

(b ) Low level degraded qum tsehfet ,(d) Medium level degraded qum tsehfet ,(f) High level degraded qum tsehfet

## 4.2 Restoring historical Amharic document images

In this study, an attempt is made to explore mathematical morphology and wavelet transform for restoring Amharic degraded document images. These restoration techniques are implemented using built-in methods in MATLAB image processing Toolbox. Historical Amharic document images containing low, medium, and high level of degradation are supplied to the module for testing. The quality of the pre-processed image was measured by the widely used image quality measurement method called Peak Signal-to-Noise Ratio (PSNR) and MSE.

### 4.2.1 Mathematical morphology

Mathematical morphology gives an approach to the processing of digital images which is based on of the object in the image shape [55]. Morphological operations rely on relative ordering of pixel values, not on their numerical values; thus, the operations are especially suited to binary image processing and it can be applied also to greyscale images such that their absolute pixel values are of no or minor interest. In this study, mathematical morphology is tested with Otsu thresholding techniques.

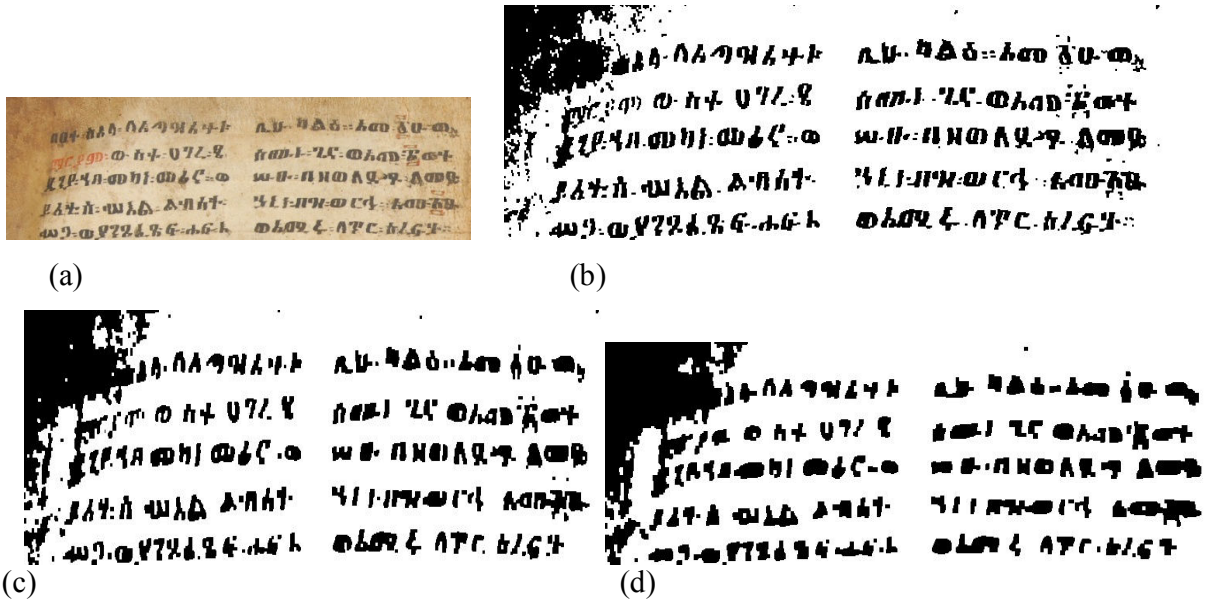
Mathematical operations probe an image with a small shape or template called a structuring, or structure element (SE). Structuring element construct by using the function **strel ( )** with variety of shapes and sizes. Its basic syntax is: - **SE= Strel (shape, parameters),** where shape is a string specifying and desired shape like “square”, “circle”, “diamond”, or “rectangle” and parameters is a list of integer that specify information about the shape.

We tested different structural element by using different parameter that applies on MATLAB function in-terms of PSNR and MSE. The reason why square is used is that, on one hand it provides equal distribution (equally affect) since the length and width are equal [56] and, on the other hand it provides the best result in-terms of PSNR and MSE than any other shapes used in the experimentation. As of the parameter used (see table 4.1 below), an integer of 2 is selected as the best since it provides the least MSE and the highest PSNR, as compared to 1, 3 and 4. Parameter of an integer 1 has no effect on the document image and the rest register less results.

When the result of the document image was visualized (see figure 4.3) the visibility of document image decreases as parameter size increased.

| Structure element | MSE | PSNR |
|---|---|---|
| **Strel (`square`,2)** | 0.0798 | 59.7644 |
| **Strel (`square`,3)** | 0.0872 | 58.9864 |
| **Strel (`square`,4)** | 0.1003 | 58.1788 |

**Table 4.1 performance of different structural element for high level degraded document**



(a) (b) (c) (d)

**Fig 4.3 sample of different structuring element**

(a) original (b) strel (`square`, 2) (c) strel (`square`, 3) (d) strel (`square`, 4)

As the results depicted in Table 4-1 shows when structuring element parameter increased from 2 to 4 the performance measured in PSNR was decreased. As Kedda[62] notes, a sequence of morphological operations using small structuring elements are computationally more efficient than one operation using a large structuring element. As a result, for experimenting the various morphological operations structural element of 2 is used.

Once the optimal structural element is identified, three mathematical morphology techniques of dilation, erosion and combination of morphology are tested with Otsu thresholding algorithms. The first morphological technique that was used under this experimentation is dilation. Dilation is a transformation that produces an image that is the same shape as the original, but is a different size. Dilation stretches or shrinks the original figure and it increases the valleys and enlarges the width of maximum regions. Dilation algorithm as the function used in MATLAB is elaborated in Listing 4.1. The dilation algorithm takes two parameters as inputs.

The first is the gray scale image by converting the true color by using rgb2gray ( ) function and the second input is set of parameter and shape known as a structuring element. It is this structuring element that determines the precise effect of the dilation on the input image using the built-in function imdilate (image, structuring element) in MATLAB Image Processing Toolbox. Finally it returns the dilated document images.

LISTING 4.1 DILATION FUNCTION IN MATLAB

```
file = strcat('HD',int2str(i),'.jpg');
df = strcat(directory, file);
    ndf = strcat(directory,'EX_DI_',file);
     GrayImage = ImageReader(df);
      se=strel('square',2);
        fe=imdilate(GrayImage,se);
          ThresholdedImage = OtsuThresholding(fe);
           report = ImageWriter(ThresholdedImage, ndf);
```

The second morphological technique that was used under this experimentation is Erosion. Erosion is used to reduce objects in the image and known that erosion reduces the peaks and enlarges the widths of minimum regions. So that, algorithm tends to remove some of the foreground (bright) pixels from the edges of regions of foreground pixels. Erosion algorithm as

the function used in MATLAB is the same as dilate function in Listing 4.1 the only difference returns the erosion of image by structuring element structure by using the built-in function imerode (shape, parameter) in MATLAB Image Processing Toolbox.

LISTING 4.2 EROSION FUNCTION IN MATLAB

```
file = strcat('HD',int2str(i),'.jpg');
df = strcat(directory, file);
    ndf = strcat(directory,'EX_DI_',file);
     GrayImage = ImageReader(df);
      se=strel('square',2);
        fe=imerode(GrayImage,se);
          ThresholdedImage = OtsuThresholding(fe);
```

As depicted below in Listing 4.3 the third technique implemented under this experimentation is combination of mathematical morphology which work on both the background as well as the foreground of a historical degraded document image. First an open and closed operation is performed to restore the background of the given document image and then the output was given to the second operation to enhance the domain of interest or the foreground of the degraded document which is selected as the best in this research based on the experiment result it give based on the well known image metrics measurement and the visualised of the resulted document (ground truth). For the second operation that apply to enhance the foreground of a given document image the structuring element of parameter 700 is set because it yields better PSNR and lower MSE than other. Finding the optimal structure element parameter was a challenge in this experimentation.

LISTING 4.3 COMBINATION OF MORPHOLOGY IN MATLAB

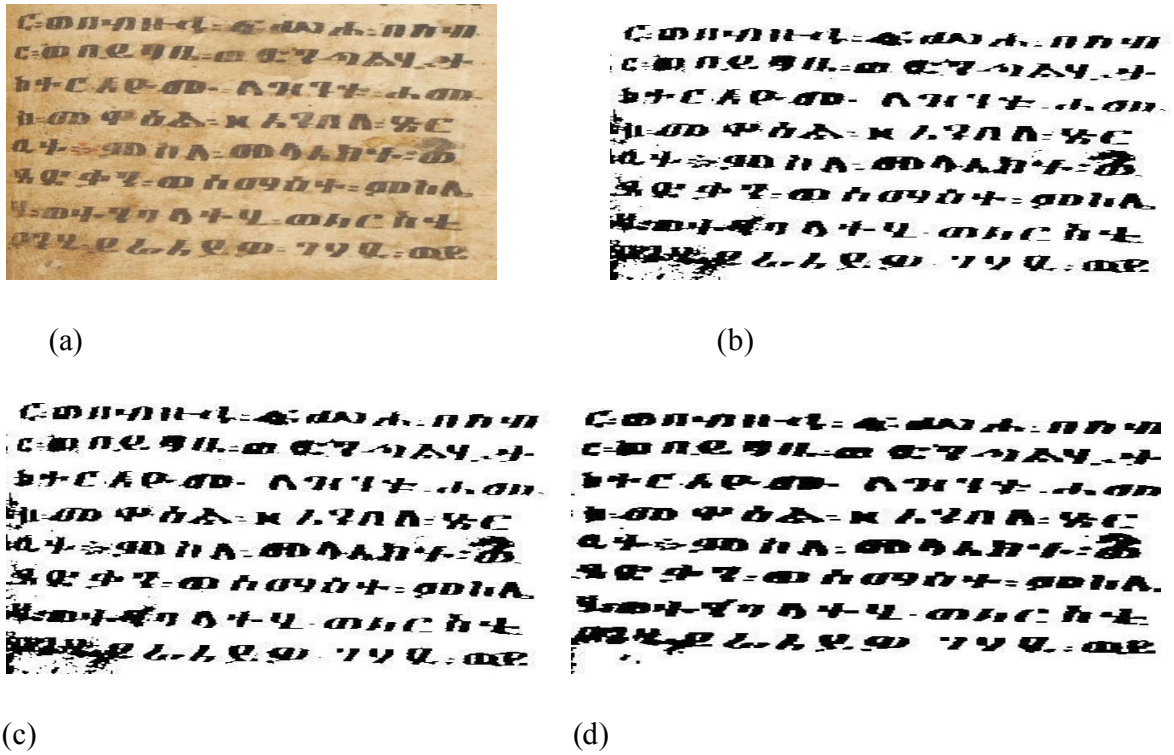
```
file = strcat('HD',int2str(i),'.jpg');
df = strcat(directory, file);
 ndf = strcat(directory,'EX_CMO_',file);
   GrayImage = ImageReader(df);
     se=strel('square',2);
       le=strel('square',700);
         l=imopen(GrayImage,se);
          fe=imclose(l,se);
            J = imsubtract(imadd(fe,imtophat(fe,le)), imbothat(fe,le));
               ThresholdedImage = OtsuThresholding(J);
                  report = ImageWriter(ThresholdedImage, ndf);
```

Table 4.2 shows summaries of performance registered by using all three morphological techniques used under this experimentation along with their MSE and PSNR for low, medium and high level of degraded historical document images.

| Degraded Level | Dilation with Otsu | | Erosion with Otsu | | Combination with Otsu | |
|---|---|---|---|---|---|---|
| | Avg.MSE | Avg.PSNR | Avg.MSE | Avg.PSNR | Avg.MSE | Avg.PSNR |
| **Low** | 0.0341 | 63.2540 | 0.0457 | 61.6487 | 0.0314 | **63.7258** |
| **Medium** | 0.0612 | 60.3069 | 0.0640 | 60.0994 | 0.0566 | **60.6322** |
| **High** | 0.0946 | 58.6299 | 0.1023 | 58.0983 | 0.0798 | **59.7644** |
| **Average** | 0.0633 | 60.7302 | 0.0706 | 59.9488 | 0.0559 | **61.3741** |

**Table 4.2 performance of Morphological Operation**

According to the performance result shown in table 4.2, for all levels of degraded document images Otsu thresholding along with combination of mathematical morphology was found best as compared with others with highest average score of PSNR **61.3741 DB** and the lowest score of MSE **0.0559.** Visualization of the result in fig 4.4 also showed that combination of morphological operations with Otsu thresholding was better than the other two in terms of readability and visibility.



(a) (b)

(c) (d)

**Fig 4.4 Sample result of high level degraded document from experiment 1**

(a) Original high level degraded image (b) Dilation with Otsu (c) Erosion with Otsu (d) Combination with Otsu

As a result, we conclude that the combination of mathematical morphology operations along with global thresholding algorithm (Otsu) performed well for historical Amharic degraded document images at various degradation levels than Dilation and Erosion with Otsu.

### 4.2.2 Wavelet transform

This experiment evaluates three wavelet transform families namely Haar, Symlet and Daubechies on restoring historical Amharic document images. The *wavelet transform* (WT) is a powerful tool of signal processing for its multi- resolution possibilities based on signal and it also include threshold of its own unlike mathematical morphology who relay on the structure and shapes.

The researcher implementation code for all three of wavelet transform is done by combining different tool box in the MATLAB. This experimentation which involves the selection of thresholding, reconstruction and restoring degraded historical document images. The implementation haar wavelet code in MATLAB is presented in listing 4.4.

The haar function first take the gray scale image as an input and returns the wavelet decomposition of the input image matrix at level N of decomposition, using the wavelet named in string 'db1' or haar in this case and then reconstruct the signal by using horizontal, vertical, approximate, diagonal detail of an image as well as the thresholding value.

Among the different families of wavelets three of them namely haar, dabenchies and symlets are used. The reason we selected these wavelets are because they are very widely used in digital image processing for restoring images [57] – [61].

LISTING 4.4 HAAR WAVELET IN MATLAB

```
[C,S] = wavedec2(image,N,'db1');.

cA1 = appcoef2(C,S,'db1',1);

[cH1,cV1,cD1] = detcoef2('all',C,S,1);

 cA2 = appcoef2(C,S,'db1',2);

[cH2,cV2,cD2] = detcoef2('all',C,S,2);

For i=1 : 2

  Ai = wrcoef2('a',C,S,'db1',i);

   Hi = wrcoef2('h',C,S,'db1',i);

   Di = wrcoef2('d',C,S,'db1',i);

   Vi = wrcoef2('v',C,S,'db1',i);

end

   [thr,sorh,keepapp] = ddencmp('den','wv',X);

   disp('global positive threshold value is');

    disp(thr);

[XC,CXC,LXC,PERF0,PERFL2]= wdencmp
('gbl',C,S,'db1',N,thr,'h',keepapp);

       for iLevel = N:-1:1,

           M=waverec2(CXC,LXC,'db1');

           out1=uint8(M);

        end
```

The second wavelet technique used under this experimentation is Daubechies Wavelets. Daubechies wavelets are a family of orthogonal wavelets and they are characterized by a maximal number of vanishing moments for some given support width [60][61]. The implementation of daubechies wavelet code in MATLAB is some what similar with haar. The only difference is that instead of using **"db1"** built in wavelet function shown in listing 4.4 we used **"db4"** built in wavelet function in here. The whole code is annexed in appendix II.

The Daubechies function first take the gray scale image as an input and then it convert the image returns the wavelet decomposition of the input image matrix at level N of decomposition, using the wavelet named in string 'dbN' where N is greater than 1. In this case and then reconstruct the signal by using horizontal, vertical, approximate, diagonal detail of an image of the image as well as the thresholding value.

The third implemented wavelet transform algorithm is Symlet. The Symlets are nearly symmetrical wavelets proposed by Daubechies as modifications to the db family. The properties of the two wavelet families are similar. This wavelet represents the same wavelet as `*symN*`. In *symN*, *N* is the order. Some authors use 2*N* instead of *N [60]*.

The Symlet function first take the gray scale image as an input and then returns the wavelet decomposition of the input image matrix at level N of decomposition, using the wavelet named in string 'symN' where N is greater than 1 in this case and then reconstruct the signal by using horizontal, vertical, approximate, diagonal detail of an image as well as the thresholding value. The implementation of symlet wavelet code in MATLAB is the same as the other two the only difference is that instead of using **"db1"** function shown in listing 4.4 and **"db4"** function for daubechies we used "**sym**" function here. The whole code is annexed in appendix II.

Table 4.3 shows summaries of performance registered by using all three wavelet techniques used under this experimentation along with their MSE and PSNR for low, medium and high level of degraded historical document images.

| **Degraded level** | **Haar wavelet** | | **Daubechies wavelets** | | **Symlet wavelet** | |
|---|---|---|---|---|---|---|
| | Avg.MSE | Avg.PSNR | Avg.MSE | Avg.PSNR | Avg.MSE | Avg.PSNR |
| **Low** | 0.0341 | 63.3416 | 0.0341 | **63.3474** | 0.0341 | 63.3468 |
| **Medium** | 0.0608 | **60.3325** | 0.0608 | 60.3311 | 0.0608 | 60.3300 |
| **High** | 0.0820 | 59.7841 | 0.0819 | **59.7910** | 0.0821 | 59.745 |
| **Average** | 0.0589 | 61.1527 | 0.0589 | 61.1565 | 0.059 | 61.1406 |

**Table 4.3: Performance of Wavelet Transform**

From the second experimentation, it was observed that for high-level degraded and low-level degraded document images, the **Daubechies wavelets** was found better than the other two wavelets with an average PSNR of **63.3474 DB** for low level and **59.7910 DB** for high level degraded document images. But, for medium-level degraded document image haar wavelet was better than others of two techniques by scoring an average PSNR of **60.3325 DB**. The reason haar was performing better than the other two for medium level degraded document is that haar is discontinuous, and resembles a step function (work either on high (1) or low (0) frequencies).

**Fig 4.5 Sample result of medium level degraded document from experiment 2**

(a) **original medium level degraded document image (b) result of Daubechies (c) result of Haar (d) result of Symlet**

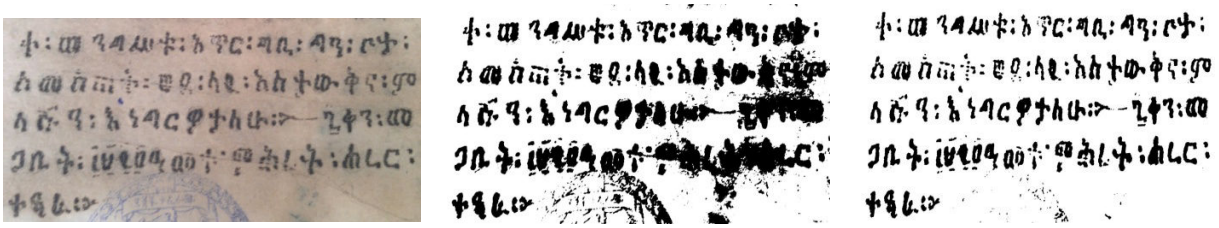
However, as depicted in Figure 4.6 the visualization of the sample example result images of medium level degraded document image we observed that, in the result of haar (c) the document images results have shown some characters were broken and less bright and in the result of Symlet (d) the images are less bright and some characters tend to fade. In addition to that on average daubechies scored highest than others by **61.1565 DB**. As a result, from wavelet transform the daubechies wavelet is selected.

Table 4.4 summarizes all the best results from all experimentations conducted in this study. All the codes that are written for all experimentation are shown in appendixes I and II.

| Degraded level | Experimentation 1 | Experimentation 2 |
|---|---|---|
| | ***i.*** **Combination of Morphological operation with Otsu** | ***ii*** **Dabechies wavelet** |
| **Low** | 63.7258 | 63.3474 |
| **Medium** | 60.6322 | 60.3311 |
| **High** | 59.7644 | 59.7910 |
| **Average** | 61.3741 | 61.1565 |

**Table 4.4 summarizes all the best results from all experimentations**

As clearly shown in Table 4.4, for all levels of degradation, the step wise combination of morphological operations and Otsu thresholding registered best performance in terms of PSNR when compared with the other experiments. From over all performance of the two experimentations, the combinations of mathematical morphologies with Otsu tresholding result in **61.3741 DB** which is **0.2176 DB** greater than that of Dabechies**.**



(a) (b) (c)

**Figure 4.6: Comparison between Dabechies and combination of morphology method**

(a) An original high-level degraded document image, (b) result of dabechies method, (c) result of combination of morphology method.

In addition to this, dabechies wavelet tends to create blurred images and convert the words in the image another form while there is content in it. Figure 4.6 (a) clearly shows an original high-level degraded document image (b) result of dabechies method (c) and result of combination of morphology method.

Therefore, based on the experimentations conducted, quality historical document images is obtained by using **Combination of Morphological Operations with Otsu thresholding** ,and hence this is integrated to Amharic document image retrieval system as presented in section 4.3.

### 4.2.3 Restoring real life document

Even though it is not the main focus of this study, it would be imperative to check whether the selected technique (combination of morphology) in focus would be applicable to real life documents as well. This was done because all the data sets used by previous researchers were real life documents and the result of this research is also expected to integrate with it. In-order to test we used the Biniam's [10] real life data set and tested by using his selected noise filtering algorithms (wiener with Otsu). Table 4.5 shows a comparison of the performance result before and after integrating the selected restoration techniques with Biniam's noise removal modules.

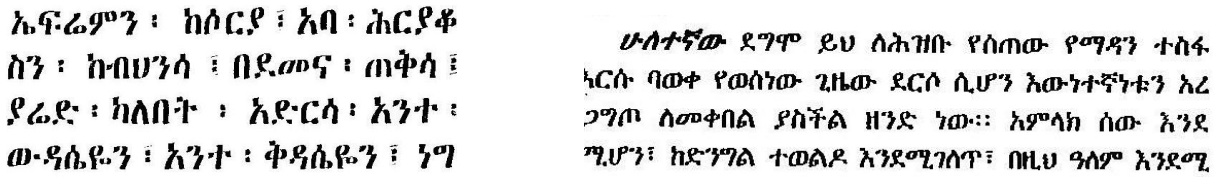


**Fig 4.6 Some of the results of restoring real life documents**

| Level of Noise on real-life | Using Biniam's selected noise filter | | After integrating restoration technique | |
|---|---|---|---|---|
| | Avg.MSE | Avg.PSNR | Avg.MSE | Avg.PSNR |
| **Low level** | 0.0148 | 66.7372 | 0.0017 | 76.2420 |
| **Medium level** | 0.0629 | 60.6378 | 0.0048 | 72.7603 |
| **High level** | 0.0868 | 58.9490 | 0.0048 | 72.0182 |
| **Average** | 0.0548 | 62.1080 | 0.0038 | 73.6735 |

**Table 4.5 Performance of the selected restoration technique on real life documents**

As depicted in table 4.5, after integrating the selected restoring technique a better result was obtained. For all the noise levels in real life documents, MSE values are decreased and PSNR values are increased. This experimentation showed us in order to improve the quality of document images; there is a need to combine noise filtering with image restoration techniques.

## 4.3 Integrating the selected Restoration Algorithms to the Amharic Document Image Retrieval System

The best restoration techniques selected in this study is integrated with an existing ADIRS developed by Java programming Language. To integrate the MATLAB code with Java, we used MATLAB Builder JA software. MATLAB® Builder TM JA enables to create Java TM classes from MATLAB® programs, which can be deployed royalty-free to desktop computers or web servers that do not have MATLAB installed [63].

The Java program below (see listing 4.7) was developed by the researchers to call the MATLAB file that is compiled as Java package. After integrating the image restoring module developed in MATLAB, we used low-level, medium-level, high-level degraded document images to evaluate the performance of the system which is measured by precision, recall and F-measure. Integrating the two programming languages was a difficult task for the researchers because of the different parameters and procedures that must be followed strictly. This has greatly consumed the time of the researchers.

LISTING 4.5 INTEGRATING THE MATLAB IMPLEMENTATION WITH JAVA

```
package ImageDocuments;
import Restoring.Restoring;
import com.mathworks.toolbox.javabuilder.MWArray;
import com.mathworks.toolbox.javabuilder.MWNumericArray;

public class ImageRestoring {
    public void ImageRestoring()
   {
       System.out.println("I'm inside the retoring method.");
      MWNumericArray n = null;   /* Stores input value */
      Object[] result = null;    /* Stores the result */
      Restoring myRestoring = null;     /* Stores ImageRestoring class instance */
      try
      {
         System.out.println("Trying to restore ...");
         myRestoring = new Restoring();

         myRestoring.Restoring();
         System.out.println("Done Image restoring ...");
      }

      catch (Exception e)
      {
         System.out.println("Exception: " + e.toString());
      }

      finally
      {
         MWArray.disposeArray(n);
         MWArray.disposeArray(result);
         if (myRestoring != null)
           {
            myRestoring.dispose();
         }
      }
```

We selected query words and tested the performance using the different degraded documents. Then, we applied restoring algorithm to the same degraded document images and produced clean images. This clean document images are tested using query words and the performance is measured again. Experimental results before and after integration of the restoration technique is shown in Tables 4.6 for low-level degraded Amharic document images.

| Query words | Before integration of proposed module | | | After integration of proposed module | | |
|---|---|---|---|---|---|---|
| | **Recall** | **Precision** | **F-measure** | **Recall** | **Precision** | **F-measure** |
| **በዜማ ደርሳል** | 100 | 100 | 100 | 100 | 100 | 100 |
| **ነጭ ዕጣን** | 100 | 100 | 100 | 100 | 100 | 100 |
| **ቅዳሴ ማርያም** | 100 | 75 | 85.71 | 100 | 100 | 100 |
| **አልጋ ወራሽ** | 0 | 0 | 0 | 100 | 75 | 85.71 |
| **በመንፈስ ቅዱስ** | 100 | 100 | 100 | 100 | 100 | 100 |
| **የአክሱም ሥልጣኔ** | 100 | 50 | 66.67 | 100 | 50 | 66.67 |
| **Average** | **83.33** | **70.83** | **75.39** | **100** | **87.5** | **92.06** |

**Table 4.6 System performance for low-level degraded document images before and after integration of the proposed module**

For document images that contain small amount of degradation, the performance of the system without the integration of the restoration module is on the average 75.39% F-measure. After restoring using combination of mathematical morphology with Otsu thresholding, the performance of the system increased on the average to 92.06% F-measure, which showed an average improvement of 16.67% F-measure.

Low-level degraded Amharic document images did not retrieved before integrating the restoration techniques because of noise characters merge with others and form another character. As a result of which, they lose their feature. However, after restoration technique module was integrated with the existing system, the system able to retrieve the degraded words; for example, as shown in Table 4.6 for query **አልጋ ወራሽ** the performance improved from 0 percent to 85.71 percent F-measure and for **ቅዳሴ ማርያም** the performance improved from 85.71 percent to 100 percent F-measure. The figure 4.7 below shows how **አልጋ ወራሽ** changes before and after applying restoration technique.

(a ) (b)

**Fig 4.7 figure 4.7 below shows how አልጋ ወራሽ changes before and after applying restoration technique**

**(a ) Before integrating applying restoration technique (b) After integrating applying restoration technique**

An experiment is also conducted to test the performance of the system on medium level degraded Amharic document images as shown in table 4.7.

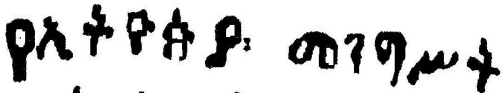
| Query words | Before integration of proposed module | | | After integration of proposed module | | |
|---|---|---|---|---|---|---|
| | Recall | Precision | F-measure | Recall | Precision | F-measure |
| የኢትዮጵያ መንግስት | 0 | 0 | 0 | 100 | 50 | 66.67 |
| ቤተ ክርስቲያን | 100 | 100 | 100 | 100 | 100 | 100 |
| ራስ ተፈሪ | 100 | 75 | 85.71 | 100 | 100 | 100 |
| እስጢፋኖስ አድርገን | 100 | 75 | 85.71 | 100 | 75 | 85.71 |
| ሰማዕታት ጋር | 100 | 75 | 85.71 | 100 | 75 | 85.71 |
| **Average** | **80** | **65** | **71.43** | **100** | **80** | **87.62** |

**Table 4.7 System performance for medium-level degraded document images before and after integration of the proposed module**

As we can see in Table 4.7, the performance of the system before image restoration techniques is on the average 71.43 % F-measure. After we integrate combination of mathematical morphology and Otsu thresholding for image restoration, the performance of the system increased on the average to 87.62 % F-measure. For medium level degradation, the retrieval system showed an average improvement of 16.19 % in F-measure.

For queries like **የኢትዮጵያ መንግስት** the existing system is not retrieving relevant document that exist with in the corpus. This is because the word image is not properly extracted from the documents because of degradation. But, after applying restoration techniques noises are greatly reduced and the system is able to identify word images. As the result, the effectiveness of the system for query word **የኢትዮጵያ መንግስት** increased from 0 % to 66.67 % F-measure. Figure 4.8 below shows how **የኢትዮጵያ መንግስት** looks like before and after applying restoration and integrated with the system.

( a)



( b)

**Figure 4.8 shows how የኢትዮጵያ መንግስት looks like before and after applying restoration and integrated with the system**.

**(a ) Before integrating applying restoration technique (b) After integrating applying restoration technique**

Further experiment is also conducted to see the performance of DIRS on high level degraded document images. Table 4.8 shows the system performance for high-level degraded document images before and after integration of the proposed module.

| Query words | Before integration of proposed module | | | After integration of proposed module | | |
|---|---|---|---|---|---|---|
| | Recall | Precision | F-measure | Recall | Precision | F-measure |
| አብያተ ክርስቲያናት | 0 | 0 | 0 | 100 | 50 | 66.67 |
| ቅርሳ ቅርስ | 100 | 100 | 100 | 100 | 100 | 100 |
| ሥነ ምግባር | 100 | 33.33 | 49.99 | 100 | 67 | 80.24 |
| በአገር አቀፍ | 100 | 100 | 100 | 100 | 100 | 100 |
| ጎብረ ብሔር | 0 | 0 | 0 | 75 | 50 | 60 |
| **Average** | **60** | **46.67** | **49.99** | **95** | **73.4** | **81.38** |

**Table 4.8 System performance for high-level degraded document images before and after integration of the proposed module**

For high degraded document images, the performance increased on the average from 49.99 % F-measure before integrating the proposed restoration technique to 82.72 % F-measure after the integration. This is due to the fact that the restoring method is effective in restoring the degraded document image that is prevalent in historical Amharic document images.

For queries like **አብያተ ክርስቲያናት** and **ንብረ ብሔር** the existing system is not retrieving relevant document that exist with in the corpus. This is because the word image is not properly extracted and some of characters are broken from the documents because of degradation. But, after applying restoration techniques noises are greatly reduced and the system is able to identify word images. As the result, the effectiveness of the system for query word **አብያተ ክርስቲያናት** increased from 0 % to 66.67 % F-measure and for query word **ንብረ ብሔር** increased from 0 % to 60 %. Figure 4.9 below shows how **አብያተ ክርስቲያናት** looks like before and after applying restoration and integrated with the system.

(a ) (b )

**Figure 4.9 shows how አብያተ ክርስቲያናት looks like before and after applying restoration and integrated with the system**.

**(a ) Before integrating applying restoration technique (b) After integrating applying restoration technique**

## 4.4 Searching Using a Query with N-Terms

In addition to integrating historical document image restoring technique to the previously developed Amharic DIRS, this study also aims to improve searching by accepting N - query words from the user and apply vector space model. The previous system developed by [7] [8] [9] accepts only a single query and Biniam [10] tried to improve it by accepting at most two –words query.

As can be seen in the newly proposed architecture of the Amharic DIRS in this study (see Figure 3.1).The user is able to enter a query with N- terms. Since, searching with N-term is implemented by modifying the existing vector space model, we represented documents by a vector of keywords extracted from the document, with associated weights representing the importance of the keywords in the document and within the whole document collection; likewise, a query with n terms are modeled as a list of keywords with associated weights representing the importance of the keywords in the given query. Once the term weights are determined, we used vector space model to match between the query and documents. Finally, the relevant documents are retrieved to the user in a ranked order.

As presented in table 4.9, we experimented on a clean, low, medium, high level degraded dataset that has been restored by using combination of mathematical morphology and Otsu thresholding.

| Query with N-terms | Recall | Precision | F-measure |
|---|---|---|---|
| ቅዱስ | 100 | 100 | 100 |
| የአክሱም ግዛት | 100 | 100 | 100 |
| እንግዲ በዘህ መስፈርት | 100 | 66.67 | 80 |
| አክሱም በውጭ ኃይሎች ተጽዕኖ | 100 | 75 | 85.71 |
| የታሪክ ሊደት ሆነና በመጀመሪያው መቶ ክፍለ ዘመን | 100 | 50 | 66.67 |
| የግል ቤት | 100 | 75 | 85.71 |
| አምላክ ወልደ አምላክ እግዚአብሔር | 100 | 66.67 | 80 |
| **Average** | **100** | **76.19** | **85.44** |

**Table 4.9 Result of experimenting Query with N-terms Retrieval**

According to the result of the experimentation shown in Table 4.9, the average value of recall, precision and F-measure measures are 100%, 76.19% and 85.44% respectively. From the experimental result we observed that as the number of document images increases the performance of the DIR system decreases. This is because for some document images the word images was not extracted well.

## 4.5 Integrating Image Viewer and Enhancing the interface of ADIRS

The third and final contribution of this study is integrating the image viewer and improving the current interface of Amharic DIR.

The previous system had a label (box) for the user to enter the query words. The query is rendered (convert to image), features extracted, similarity measured between query and document feature vectors, and finally the relevant documents are presented to the user in a ranked order. But the documents retrieved are on the java console. As a result of that we tried to integrate image viewer and improve the interface.

We designed a graphical user interface (GUI) to increase the usability and friendly experience of the user with ADIRS. First, the user enters a query with n-terms and the system searches for relevant documents .Then, the result are displayed in the area dedicated for the purpose. To increase the user experience we also provided that every returned result to have a link so that a user simply clicks on one of it to view the document image itself. That means, every returned document are linkable so that when the end user click on one of the retrieved document it opened for them easily and in a user friendly manner as presented in Fig 4.10.

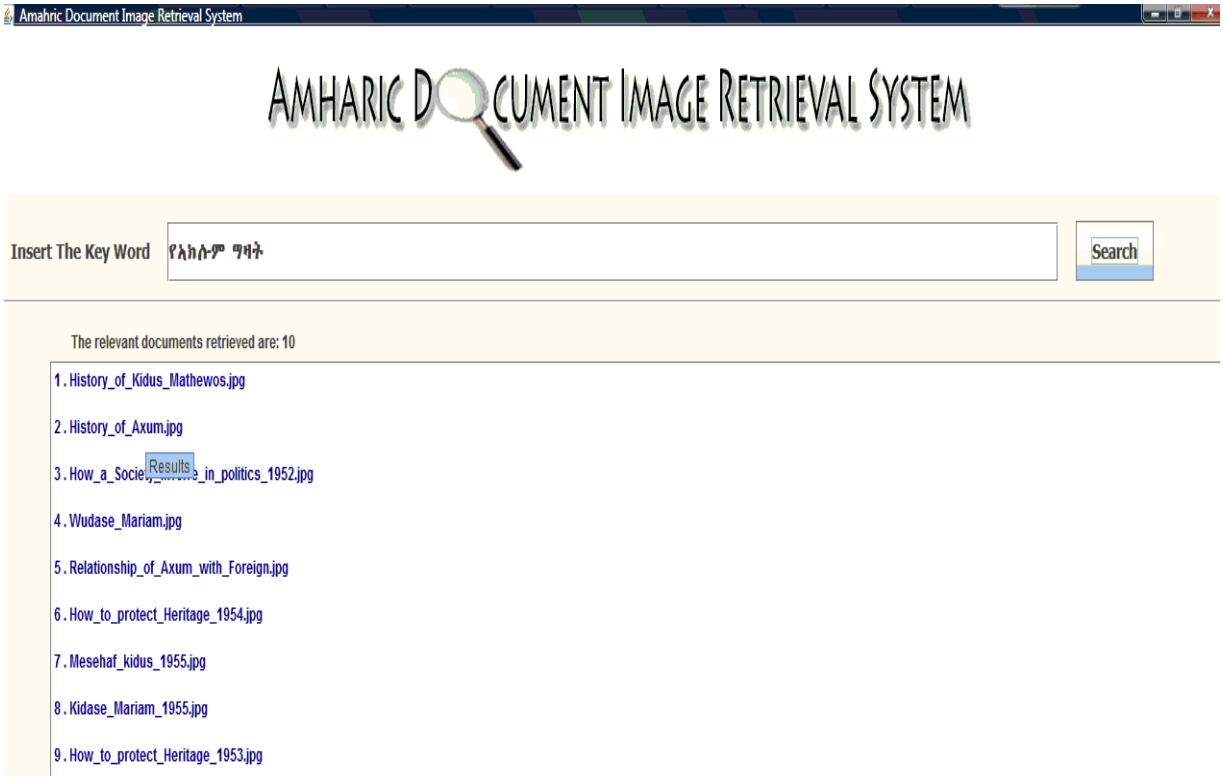



**Fig 4.10 The user interface designed in this study**

## 4.6 Finding and Challenges

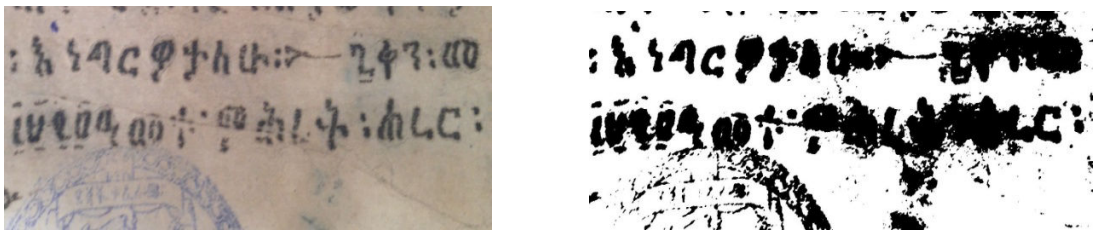
In this research, we attempted to integrate historical document image restoring technique, image viewer for enhancing the user interface as well as improve searching by supplying N- word queries to the Amharic DIR system. This study investigates three mathematical morphology operators with Otsu thresholding and three wavelet transform techniques. The experimental results showed that the combination mathematical morphology with Otsu threshold technique was best to restore low, medium and high -level degraded Amharic Document Images with PSNR of **63.7258 dB, 60.6322 dB** and **59.7644 dB** respectively**.**

Moreover, the proposed restoration techniques modules were integrated with the previous Amharic Document Image Retrieval System and tested on low-level, medium-level and high-level of degraded historical Amharic document images according to selected of restoration techniques for degraded levels. Overall after the proposed restoration techniques modules are

integrated with the previous system, the performance increased on the average by **16.67%** F-measure for low-level, **16.19 %** F-measure for medium-level, and **32.73%** F-measure for high-level degraded document images. On the average, **87.02%** F-measure is achieved for all levels of degraded document images and searching with N-word queries achieved an average performance of **85.44%** F-measure.

This shows that a better performance is registered than the previously developed system. Moreover, we tried to develop a hybrid algorithm by combining mathematical morphologies to work on historical Amharic document images. As the result, the best is obtained in terms of PSNR and image quality. Furthermore, it is chosen in this study to be the best technique of restoring degraded historical Amharic documents.

The main challenge that affect the performance of Amharic DIRS is when restoring a high level degraded historical documents especially collection of old letter that are written in "qum tsehfet" the restoring algorithms tend to convert some character into other form of shape as depicted in figure 4.11. This makes difficult for segmentation and indexing.



(a) (b)

**Fig 4.11 Example showing how the character in the original image document changed**

**(a) Original high level degraded image (b) after applying restoration technique**

Additional challenge in this study is related to indexing document images. The indexing take longer time in order to index a given document image especially when the number of document image is increased it may take 6 hours up to 2 days.

# CHAPTER FIVE

# CONCLUSION AND RECOMMENDATION

Historical Amharic documents images contain vital information regarding the past social, political, cultural, economic and other aspects. To make these documents accessible and searchable by the user, there is a need to design document retrieval system for Amharic language.

This research is a continuation of other attempts to develop a retrieval system from historical Amharic document images. Here, we developed and integrated an image pre-processing specifically restoring techniques and it also enhances usability of the interface of ADIRS, as well as the functionality of vector space model by accepting n-words query.

## 5.1 Conclusion

The main objective of this research is to integrate an image pre-processing technique for restoring historical Amharic document images. To this end, two image restoring techniques : mathematical morphology and wavelet transform are experimented. In the first experimentation three mathematical morphology namely erode, dilate and combination of morphology with Otsu thresholding were tested and in the second experimentation three wavelet techniques namely Haar, Dabechies and Symlets were tested on Amharic historical documents image. A series of experiments were conducted to select the optimal restoring techniques on low, medium and high level degraded document images.

The performance results obtained show that the combination of mathematical morphology and Otsu thresholding achieved the highest peak signal-to-noise ratio (PSNR) of 63.7258 dB for low level, 60.6322 dB for medium level and 59.7644 dB for high level degraded historical Amharic document images. This restoration technique is integrated to Amharic document image retrieval system. Then, the performance of the system is measured before and after integration of the module separately using the four levels of noise. This shows a great improvement of retrieval

effectiveness in all the degradation levels. For low level degraded document images, the performance increased from 75.39% to 92.06% F-measure. Correspondingly, for medium level degraded, it increased from 71.43% to 87.62% and for high level degraded from 49.99% to 81.38% F-measure.

This study also attempts to improve the on-line searching capability of the Amharic DIRS by adopting vector space model such that users enter their information need with N-words query and retrieve relevant documents in ranked order. Accordingly, the system able to retrieves with 80.76 % F-measure when tested on historical Amharic document images.

However, the performance of the system is affected as the level of degradation of documents increases especially documents with high level degraded that are written in qum tsehfet. In addition to that since there is unavailability of standardized corpus, the dataset contains limited number of historical and real life document images. Furthermore, the indexing take longer time in order to index a given document image especially when the number of document image is increased.

## 5.2 Recommendation

The current research enhanced the effectiveness of the previous systems by integrating restoring techniques. But, in order to increase and further improve the performance of both Amharic Document Image Retrieval system and the restoration techniques the following issues must be considered and addressed in future work.

- To enhance the performance of Amharic DIR system, an advanced restoration technique that can recover the high level degraded document images should be considered.

- Since there is unavailability of standardized corpus and the dataset contains limited number of historical document images in this research, there is a need to prepare standardized for performance evaluation in the future..

- To improve the performance of the system, an advanced indexing and document ranking algorithm that has better structure and optimization should be developed.

- Historical documents that are skewed are not considered in this research. Hence, in order to come up with a practical Amharic document image retrieval system, image pre-processing tasks such as skew detection and correction are vital and should be considered.

- The current graphical user interface of the ADIRS needs to be more advanced and more user friendly by making it web based system so that it will be accessible anywhere anytime.

## 6. References


1. P. Sankar, V. Ambati, L. Hari, and C.V Jawahar, "*Digitizing a million books: Challenges for document analysis*", Proceedings of *DAS-2006*, pages 425–436, 2006.
2. J. Banerjee, A. M. Namboodiri, and C.V Jawahar, "*Contextual restoration of severely degraded document images*," in Computer Vision and Pattern Recognition, CVPR-2009,IEEE Conference, pages 517-524,2009
3. P. Sarkar, H. Baird, and X. Zhang, "*Training on severely degraded text-line images*", Proceedings of the International Conference on Document Analysis and Recognition, pages 38–43, 2003.
4. National African Language Resource Center (NALRC), "*Ethiopia: Language & Culture*," University of Wisconsin, Madison, pages 1-2, 2010.
5. Wondewosen Mulugeta ,"*OCR For Special Type of Hand Written Amharic Text*", MSc Thesis, Department of Information Science, Addis Ababa University, Ethiopia, 2004.
6. Million Meshesha, "*Recognition and Retrieval from Document Image Collections*," PhD Dissertation, International Institute of Information Technology, India, 2008.
7. Mesfin Worku, "*Amharic Document Image Retrieval without Explicit Recognition,*" M.Sc Thesis, School of information science, Addis Ababa University, Addis Ababa, Ethiopia,2009.
8. Abreham Gebretsadik ," *Searching in Amharic Document Image Corpus",* M.Sc Thesis, School of information science ,Addis Ababa University, Addis Ababa, Ethiopia,2010.
9. Adane Letta, "*Feature Extraction and Matching In Amharic Document Image Collections*," M.Sc. Thesis, School of Information Science, Addis Ababa University, Addis Ababa, Ethiopia, 2011.

10. Biniam Asnake, "*Retrieval from real-life Amharic Document Images*," M.Sc. Thesis, School of Information Science, Addis Ababa University, Addis Ababa, Ethiopia, 2012.

11. Kibrom Tadesse, "*Towards Segmentation of Real-life Documents for Amharic Document Image Retrieval,"* M.Sc. Thesis, School of Information Science, Addis Ababa University, Addis Ababa, Ethiopia, 2013.

12. Gedion Assefa, "page segmentation in Amharic document image collection," M.Sc. Thesis, School of Information Science, Addis Ababa University, Addis Ababa, Ethiopia, 2013

13. C.R. Kothari, "*Research Methodology: Methods and Techniques"* 2nd edition,,India, New Age International Publishers, 2004

14. James Gosling and Henry McGilton," *The Java™ Language Environment*," Sun Microsystems, 2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A, 1997

15. Beaza-Yates Ricardo and Berthier Ribeiro-Neto, "*Modern Information Retrieval*", A Division of the Association for Computing Machinery: Addison-Wesley, ACM Press, 1999 .

*16*. Chew Lim Tan, Weihua Huang, Zhaohui Yu and Yi Xu ,"*Imaged Document Text Retrieval Without OCR*,IEEE Transaction on Pattern Analysis and Machine Intelligence, vol.24, Issue 6, pages 838-844, 2002.

17. Kareem Darwish and Ossama Emam ,"*Retrieving Arabic Printed Document; A Survey*", IBM Technology Development Centre, Cairo Egypt, 2006.

*18*. Konstantinos Zagoris, Kavallieratou Ergina and Nikos Papamarkos ,"*A Document Image Retrieval Syste*m", Engineering Applications of Artificial Intelligence, Vol. 23, Issue 6, pages 872-879, 2010.

19. Wondewosen Mulugeta, "*OCR For Special Type of Hand Written Amharic Text*", MSc Thesis, Department of Information Science, Addis Ababa University, Ethiopia, 2004.
20. M. B Kokare and M. S Shirdhonkar, "*Document Image Retrieval*: *An Overview*", International Journal of Computer Application,Vol. 1,Issue 7, pages 114-119, 2010.
21. Million Meshesha and C.V Jawahar "*Matching Word Images for Content-Base Retrieval from Printed Document Images*", International Journal on Document Analysis and Recognition,Vol. 11,Issue1,pages 29-38, 2008.
22. J. Sauvola, T. Seppanen, S. Haapakoski and M. Pietikainen, "*Adaptive Document Binarization,*" in Proc. 4th Int. Conf. On Document Analysis and Recognition, Ulm, Germany, pages147-152, 1997.
23. R. Kasturi, L. O'gorman and V. Govindaraju, "*Document Image Analysis: A primer*," *Sadhana*, vol. 27, no.1, pages 3–22 , 2002.
24. Soda Marinai , "*Tools for Document Image Retrieval in Digital Libraries: the AIDI System* ", Department of System Informatics University of Florence, Italy, 2009
25. Lawrence O'Gorman and Rangachar Kasturi,"*Document image analysis",* IEEE Computer Society Executive Briefings,2009.
26. T.M. Ha and H. Bunke, "*Image Processing Methods for Document Image Analysis*," Handbook of Character Recognition and Document Image Analysis, pages 1-47, 1997.
27. Pedro Daniel ,da Rocha Melo and e Castro, "*Restoration of poor documents*", International Journal of Computer Application, Vol. 1,Issue 7, pages 114-119, 2007
28. Hamid Rahimizadeh, M.H Marhaban, R.M Kamil and N.B Ismail, "*Color Image Segmentation Based on Bayesian Theorem and Kernel Density Estimation*", European Journal of Scientific Research, Vol.26,Issue 3, pages 430-436, 2009
29. Mandl T, "R*ecent Developments in the Evaluation of Information Retrieval Systems*", Moving Towards Diversity and Practical Relevance, Information Science, University of Hildesheim , 2007.
30. Manesh B. Kokare and Shirdhonkar M. S," *Document Image Retrieval*: *An Overview"*, International Journal of Computer Application. Vol 1, Issue 7,pages 114-119, 2010.
31. Ryszard S. Chora´s ,"*Image Feature Extraction Techniques and Their Applications for CBIR and Biometrics Systems*", International Journal of Biology and Biomedical Engineering, Vol. 1,Issue 1, pages 6-16, 2007.

32. Salton and McGill," *Introduction to modern information retrieval*", McGram Hill Book Company, 1983
33. Punitha P., Naveen and Guru D.S ,"*Indexing and Retrieval of Document Images by Spatial Reasoning*" ,In Proceedings of the Third International Conference on Distributed Computing and Internet Technology (ICDCIT'2006), Bhubaneswar, India, LNCS 4317, pages 457 – 464, 2006.
34. C.D. Manning, P. Raghavan and H. Schutze, "*Introduction to information retrieval"*, USA: Cambridge university press, 2008
35. H. S. Baird,"*Document image defect models and their uses*", In *ICDAR '93*, Tsukuba, Japan, pages 62-67,1993.
36. K. Kise and D. Doermann. "*Second International Workshop on Camera-Based Document Analysis and Recognition"*,2007.
37. S. Rice, G. Nagy, and T. Nartker, "*Optical Character Recognition", An Illustrated Guide to the Frontier*, Kluwer, 1999.
38. H. Baird,"*The state of the art of document image degradation modeling*", In IAPR International Workshop on Document Analysis Systems, pages 1–16, 2000.
39. F. Drira,"*Towards restoring historic documents degraded over time*", In Document Image Analysis for Libraries, pages 350–357, 2006.
40. F. Stanco and G. Ramponi "*Detection of Water Blotches in Antique Documents*", In Proc. 8th COST 276 Workshop, 2005.
41. Tessema, Mindaye, Meron Sahlemariam and Teshome Kassie, "*The Need for Amharic WordNet*",Computer Science Department, Addis Ababa University, IS Division, UN ECA, Ministry Finance and Economic, 2010 .
42. Bender, M.L, Sydney W. Head, and Roger Cowley, "*The Ethiopian writing System*", Language in Ethiopia, London: Oxford University Press, 1976
43. Bethlehem Mengistu , "*N-Gram-Based Automatic Indexig For Amharic Text*", MSc Thesis, Department of Information Science, Addis Ababa University, Ethiopia, 2002
44. LESLAU, W, "*Introductory Grammar of Amharic",* Introductory Grammar of Amharic By Porta Linguarum Orientalium, 2000.
45. Encarta, M, *Encyclopedia*, Retrieved June 12, 2008, from African_Languages: http://encarta.msn.com/encyclopedia_761565449/African_Languages.html#s9 , 2007.

46. Wikipedia , *Amharic*, Retrieved June 19, from wikipedia: http://en.wikipedia.org/wiki/Amharic,2009.
47. Yimam ," የ ዓማርኛ ሰዋሰው",Addis Ababa ,ት.መ.ማ.ማ.ድ,1986.
48. B.Gangamma and Srikanta Murthy K ,"*Restoration of degraded historical document mage*", Journal of Emerging Trends in Computing and Information Sciences, vol. 3, 2012.
49. Anna Tonazzini, Luigi Bedini, Emanuele Salerno," *Independent component analysis for document restoration*", international journal on document analysis and recognition, Vol.11, pages 57-62,2004.
50. MathWorks, "*MATLAB Image Processing Toolbox 7.0 User's Guide*," The MATHWORKS Inc, 2012, Available at http://www.matlab.com, [Accessed on: 6/3/2012].
51. C. Torrence, *"Image Processing IDL Version 7.1*, 1st ed.", ITT Visual Information Solutions Inc., 2009.
52. M. Delalandre. (2010),"*Performance Evaluation of Document Image Analysis Systems*": A Primer [Online] Available: http://mathieu.delalandre.free.fr/teachings/evaluation/, [Access Date: 4/1/2012].
53. Tamrat Delessa, "*Restoring of historical Amharic document image to enhance the performance of Amharic document image retrieval system* ," M.Sc. Thesis, School of Computer Science, Gonder University, Addis Ababa, Ethiopia, 2013.
54. Fasil Fanta, "*Towards Segmentation of Real-life Documents for Amharic Document Image Retrieval*," MSc, Computer Science, University of Gondar, Ethiopia, 2012.
55. Rafel C.Gonzalez and E.wood ,"*Digital Image Processing* ",PHI 2nd Edition 2005.
56. Rafel C.Gonzalez and E.wood ,"*Digital Image Processing using MATLAB* ",Pearson Education, 2006.
57. S. G. Mallat, "*A theory for multiresolution signal decomposition*: *the wavelet representation*," vol. 11, pages 674-693, July 1989.
58. Y.-S. Zhang, "*Multiresolution analysis for image by generalized 2-d wavelets,"* MSc thesis, computer science, University of corolado, 2008.
59. S. Mallat, "*A Wavelet Tour of Signal Processing, 3rd ed., Third Edition: The Sparse Way"*, Academic Press, 3rd edition, 2008.

60. I. Daubechies, "*Ten lectures on wavelets, vol. 61 of CBMS-NSF Regional Conference Series in Applied Mathematics*", Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 1992

61. I. Daubechies, "*Orthonormal bases of compactly supported wavelets :Communications on Pure and Applied Mathematics*", vol. 41, Issue 7, pages 909-996, 1988

62. A. Ledda, "*Mathematische morfologie in de beeldverwerking Mathematical Morphology in Image Processing*," 2007.

63. G. N. Sarage and S. S. Jambhorkar, "*Noise Removal from Mammographic Image based on Mean and Median Filtering Techniques*," International Journal of Advanced Research in Computer Science, vol. 2, Issue 4, pages 498-500, 2011.

64. F. Drira, F. LeBourgeois, and H. Emptoz, "*A new PDE-based approach for singularity-preserving regularization: application to degraded characters restoration*," International Journal on Document Analysis and Recognition, pages 1-30, 2011.

65. Encyclopædia Britannica, "*Ethiopia*," Encyclopaedia Britannica Ultimate Reference Suite, Chicago: Encyclopædia Britannica, 2010.

66. L. M. Paul, "*Ethnologue: Languages of the World*, 16th edition". Dallas, Texas: SIL International, 2009.

67. A. Wimsatt and R. Wynn, "*Amharic Language and Culture Manual*," Texas State University, Texas, 2011.

68. S. Uhlig, "*Encyclopedia Aethiopica,* 1st ed.", Wiesbaden: Harrassowitz Verlag, 2003.

69. Tilahun Yeshambel, "*Amharic Document Image Retrieval Using Lingustic Features*," MSc Thesis, Department of Computer Science, Addis Ababa University, Addis Ababa, Ethiopia, 2011.

70. National African Language Resource Center (NALRC), "*Ethiopia: Language & Culture*" University of Wisconsin, Madison, pages. 1-2, 2010.

71. J. Sauvola, T. Seppanen, S. Haapakoski and M. Pietikainen, "*Adaptive Document Binarization*," in Proc. 4th Int. Conf. On Document Analysis and Recognition, Ulm, Germany, pages147-152, 1997.

72. M.C. Motwani, M.C. Gadiya, R.C. Motwani, and F.C. Harris, Jr., "*Survey of Image Denoising Techniques*," in Proc. of GSPx, Santa Clara Convention Center, Santa Clara: CA, pages. 25-37, 2004

73. K. Zagoris, K. Ergina, N. Papamarkos, "*A Document Image Retrieval System*", Engineering Applications of Artificial Intelligence, vol. 3, Issue 2, pages.872-879,2010.

74. Y. Lu and C.L. Tan, "*Information Retrieval in Document Image Databases*," IEEE Transaction on Knowledge and Data Engineering, vol. 16, Issue11, pages 42-49, 2004.

75. C. L. Tan, W. Huang, Z. Yu and Y. Xu, "*Imaged Document Text Retrieval Without OCR*," Institute of Electrical & Electronic Engineers (IEEE) Transaction on Pattern Analysis and Machine Intelligence, vol. 24, Issue 6, pages 838-844, 2004.

76. Dereje Teferi, "*Optical Character Recognition of Typewritten Amharic Character*" M.Sc. Thesis, School of Information Studies for Africa, Addis Ababa University, Addis Ababa, Ethiopia, 1999.

77. Nigussie Tadesse, "*Handwritten Amharic Text Recognition Applied to the Processing of Bank Checks*," M.Sc. Thesis, School of Information Studies for Africa, Addis Ababa University, Addis Ababa, Ethiopia, 2000.

78. Mesay Hailemariam, "*Line Fitting To Amharic OCR: The Case of Postal Address*," M.Sc Thesis, School of Information Studies for Africa, Addis Ababa University, Addis Ababa, Ethiopia, 2003.

79. S. Akram, Dr. M.U. Dar, and A. Quyoum, "*Document Image Processing - A Review*", International Journal of Computer *Applications*, vol. 10, Issue 5, pages 234-243, 2010.

80. Anita pathak, "*Restoration of documents with show through distortion*", M.sc.Thesis, Faculity of Administration,University of Ottawa,Ottawa, Canada,2000.

# APPENDIX I: MATLAB CODE FOR EXPERIMENTATION 1

## DILATION FUNCTION IN MATLAB

```
Function EX_dilate
% clear the screen
clc
% close all prevoiusly opened windows
close all
% number of files
NOF = 12;
% for loop for iterating the whole document image until number of files
for i=1:NOF
% source of input document images
directory = 'E:\final expermintation hou pc 2\data set final\high\';
% read the input document images
file = strcat('HD',int2str(i),'.jpg');
% join the directory with the filename
df = strcat(directory, file);
% the output(result) for input document images
ndf = strcat(directory,'EX_DI_',file);
% convert the true color(RGB) to grayacale  and make ready for dilate
GrayImage = ImageReader(df);
% applying structuring element
se=strel('square',2);
%perform dilate function  on gray image
fe=imdilate(GrayImage,se);
% calling otsu thresholding function
```

```
ThresholdedImage = OtsuThresholding(fe);
% write the result (dilate) document image on the file
report = ImageWriter(ThresholdedImage, ndf);
%  end of for loop
end
```

---

## EROSION FUNCTION IN MATLAB

```
Function EX_Erode
% clear the screen
clc
% close all prevoiusly opened windows
close all
% number of files
NOF = 12;
% for loop for iterating the whole document image until number of files
for i=1:NOF
% source of input document images
directory = 'E:\final expermintation hou pc 2\data set final\high\';
% read the input document images
file = strcat('HD',int2str(i),'.jpg');
% join the directory with the filename
df = strcat(directory, file);
% the output(result) for input document images
ndf = strcat(directory,'EX_ER_',file);
% convert the true color(RGB) to grayacale  and make ready for erode
GrayImage = ImageReader(df);
% applying structuring element
```

```
se=strel('square',2);
%perform erode function  on gray image
fe=imerode(GrayImage,se);
% calling otsu thresholding function
ThresholdedImage = OtsuThresholding(fe);
% write the result (erodede) document image on the file
report = ImageWriter(ThresholdedImage, ndf);
%  end of for loop
end
```

---

## COMBINATION OF MORPHOLOGY IN MATLAB

```
Function EX_combinationmor
% clear the screen
clc
% close all prevoiusly opened windows
close all
% number of files
NOF = 12;
% for loop for iterating the whole document image until number of files
for i=1:NOF
% source of input document images
directory = 'E:\final expermintation hou pc 2\data set final\high\';
% read the input document images
file = strcat('HD',int2str(i),'.jpg');
% join the directory with the filename
df = strcat(directory, file);
% the output(result) for input document images
```

```
ndf = strcat(directory,'EX_CMO_',file);
% convert the true color(RGB) to grayacale  and make ready for erode
GrayImage = ImageReader(df);
% applying structuring element on the background an image
se=strel('square',2);
% applying structuring element on the foreground of an image
le=strel('square',700);
% perform an open operation
l=imopen(GrayImage,se);
% perform a closed operation on the output of open
fe=imclose(l,se);
% perform a foreground operation by taking an output from closed
J = imsubtract(imadd(fe,imtophat(fe,le)), imbothat(fe,le));
% calling otsu thresholding function
ThresholdedImage = OtsuThresholding(J);
% write the output on the file
report = ImageWriter(ThresholdedImage, ndf);
%  end of for loop
End
```

==========

# APPENDIX II: MATLAB CODE FOR EXPERIMENTATION 2

HAAR WAVELET IN MATLAB

```
Function EX_haar
  X = ImageReader(df);  % read the input file name
%Set the level for decomposition. Compute a 2-level decomposition of the image.
```

```
N=2;
 [C,S] = wavedec2(X,N,'db1');
%Extract the level 1 coefficients.
cA1 = appcoef2(C,S,'db1',1);
 [cH1,cV1,cD1] = detcoef2('all',C,S,1);
%Extract the level 2 coefficients.
cA2 = appcoef2(C,S,'db1',2);
[cH2,cV2,cD2] = detcoef2('all',C,S,2);
%Here are reconstructed branches.
A1 = wrcoef2('a',C,S,'db1',1);
A2 = wrcoef2('a',C,S,'db1',2);
H1 = wrcoef2('h',C,S,'db1',1);
V1 = wrcoef2('v',C,S,'db1',1);
D1 = wrcoef2('d',C,S,'db1',1);
H2 = wrcoef2('h',C,S,'db1',2);
V2 = wrcoef2('v',C,S,'db1',2);
D2 = wrcoef2('d',C,S,'db1',2);
%Set the threshold.
[thr,sorh,keepapp] = ddencmp('den','wv',X);
disp('global positive threshold value is');
disp(thr);
```

```
[XC,CXC,LXC,PERF0,PERFL2]= wdencmp('gbl',C,S,'db1',N,thr,'h',keepapp);
 %Multilevel 2-D wavelet reconstruction.
for iLevel = N:-1:1,
M=waverec2(CXC,LXC,'db1');
out1=uint8(M);
 End
```

---

## DAUBECHIES WAVELET IN MATLAB

```
Function EX_Daub
  X = ImageReader(df);  % read the input file name
 %Set the level for decomposition. Compute a 2-level decomposition of the image.
N=2;
 [C,S] = wavedec2(X,N,'db4');
%Extract the level 1 coefficients.
cA1 = appcoef2(C,S,'db4',1);
 [cH1,cV1,cD1] = detcoef2('all',C,S,1);
%Extract the level 2 coefficients.
cA2 = appcoef2(C,S,'db4',2);
[cH2,cV2,cD2] = detcoef2('all',C,S,2);
%Here are reconstructed branches.
A1 = wrcoef2('a',C,S,'db4',1);
A2 = wrcoef2('a',C,S,'db4',2);
```

```
H1 = wrcoef2('h',C,S,'db4',1);

V1 = wrcoef2('v',C,S,'db4',1);

D1 = wrcoef2('d',C,S,'db4',1);

H2 = wrcoef2('h',C,S,'db4',2);

V2 = wrcoef2('v',C,S,'db4',2);

D2 = wrcoef2('d',C,S,'db4',2);

%Set the threshold.

[thr,sorh,keepapp] = ddencmp('den','wv',X);

disp('global positive threshold value is');

disp(thr);

[XC,CXC,LXC,PERF0,PERFL2]= wdencmp('gbl',C,S,'db4',N,thr,'h',keepapp);

 %Multilevel 2-D wavelet reconstruction.

  for iLevel = N:-1:1,

M=waverec2(CXC,LXC,'db4');

out1=uint8(M);

  End
```

---

SYMLETS WAVELET IN MATLAB

```
Function EX_Sym

   X = ImageReader(df);  % read the input file name

%Set the level for decomposition. Compute a 2-level decomposition of the image.

N=2;
```

```
[C,S] = wavedec2(X,N,'sym2');
%Extract the level 1 coefficients.
cA1 = appcoef2(C,S,' sym2',1);
[cH1,cV1,cD1] = detcoef2('all',C,S,1);
%Extract the level 2 coefficients.
cA2 = appcoef2(C,S,'sym2',2);
[cH2,cV2,cD2] = detcoef2('all',C,S,2);
%Here are reconstructed branches.
A1 = wrcoef2('a',C,S,' sym2',1);
A2 = wrcoef2('a',C,S,' sym2',2);
H1 = wrcoef2('h',C,S,' sym2',1);
V1 = wrcoef2('v',C,S,' sym2',1);
D1 = wrcoef2('d',C,S,' sym2',1);
H2 = wrcoef2('h',C,S,' sym2',2);
V2 = wrcoef2('v',C,S,' sym2',2);
D2 = wrcoef2('d',C,S,' sym2',2);
%Set the threshold.
[thr,sorh,keepapp] = ddencmp('den','wv',X);
disp('global positive threshold value is');
disp(thr);
[XC,CXC,LXC,PERF0,PERFL2]= wdencmp('gbl',C,S,' sym2',N,thr,'h',keepapp);
```

*%Multilevel 2-D wavelet reconstruction.*

**for iLevel = N:-1:1,**

**M=waverec2(CXC,LXC,' sym2');**

**out1=uint8(M);**

**End**

---

# APPENDIX III: JAVA CODE FOR N-QUERY TERM SEARCH

## N-QUERY TERM SEARCH

```
package ImageDocuments;
/*
 * Author: Biruk Mengistu
 */

import java.io.BufferedWriter.*;
import javax.media.jai.*;
import java.awt.image.BufferedImage.*;
import javax.media.jai.PlanarImage;
import javax.swing.JOptionPane.*;
import javax.imageio.*;
import java.io.BufferedWriter.*;
import java.awt.image.BufferedImage.*;
import java.awt.image.*;
import java.awt.*;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
import java.io.*;
import java.net.URI;
import javax.swing.JOptionPane.*;
import java.util.*;
import java.util.regex.*; // for Pattern
import javax.swing.*;

public class TextRendering extends javax.swing.JFrame {

    int width;
    int height;
    int nbands = 0;
    int[] pixel = new int[nbands];
    int dataStart = 0;
    int dataEnd = 0;
    int border = 0;
    int containt = 0;
    int Lnumber = 0;
    int count = 0;
    int count1 = 0;
    int wstart = 0;
    int wend = 0;
    PrintWriter pwvb;
```

```
    PrintWriter pwwb;
    PrintWriter pw;
    FileWriter fw;
    PrintWriter pwwbQ;
    PrintWriter pwvbQ;
    PrintWriter pwQ;
    PrintWriter pwwvQ;
    int wdatastart1 = 0;
    int vdatastart1 = 0;
    int wdataend1 = 0;
    int vdataend1 = 0;
    int wborderstart1 = 0;
    int vborderstart1 = 0;
    int wborderend1 = 0;
    int vborderend1 = 0;
    String line = "";
    int pixelValT = 0;
    int wordposT = 0;
    int pixelValM = 0;
    int wordposM = 0;
    int pixelValB = 0;
    int wordposB = 0;
    int pixelValL = 0;
    int wordposL = 0;
    int nbandsQ = 0;
    int w = 0;
    int h = 0;
    int hQ = 0;
    int hordataEnd = 0;
    int hordataStart = 0;
    int pixelaverage = 0;
    int pixelaverageQ = 0;
    int[] pixelQ = new int[nbandsQ];
    File fileRend = new File("newimage3.jpg");
    String TextRend = "";
    QueryAccessMethod AllQueryMethods = new QueryAccessMethod();

    /**
     * Creates new form TextRendering
     */
    public TextRendering() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    //    <editor-fold    defaultstate="collapsed"    desc="Generated    Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        jTextFieldWord = new javax.swing.JTextField();
        jLabelWord = new javax.swing.JLabel();
        jButtonWord = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jTextFieldWord.setFont(new java.awt.Font("Ge'ez-1", 0, 11)); // NOI18N
        jTextFieldWord.addActionListener(new java.awt.event.ActionListener() {
```

```
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jTextFieldWordActionPerformed(evt);
            }
        });

        jLabelWord.setBackground(new java.awt.Color(255, 255, 0));
        jLabelWord.setFont(new java.awt.Font("Times New Roman", 1, 14));
        jLabelWord.setText("Insert word To search ");

        jButtonWord.setText("Search");
        jButtonWord.addActionListener(new java.awt.event.ActionListener() {

            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButtonWordActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING).addGroup(layout.
createSequentialGroup().addContainerGap().addComponent(jLabelWord,
javax.swing.GroupLayout.PREFERRED_SIZE,                                    172,
javax.swing.GroupLayout.PREFERRED_SIZE).addPreferredGap(javax.swing.LayoutStyle.Compon
entPlacement.RELATED).addComponent(jTextFieldWord,
javax.swing.GroupLayout.PREFERRED_SIZE,                                    182,
javax.swing.GroupLayout.PREFERRED_SIZE).addContainerGap(javax.swing.GroupLayout.DEFAUL
T_SIZE,          Short.MAX_VALUE)).addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup().addContainerGap(212,
Short.MAX_VALUE).addComponent(jButtonWord).addGap(101, 101, 101)));
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING).addGroup(layout.
createSequentialGroup().addGap(21,                                          21,
21).addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING).add
Component(jLabelWord,             javax.swing.GroupLayout.DEFAULT_SIZE,             34,
Short.MAX_VALUE).addComponent(jTextFieldWord, javax.swing.GroupLayout.DEFAULT_SIZE, 34,
Short.MAX_VALUE)).addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED
).addComponent(jButtonWord).addGap(11, 11, 11)));

        pack();
    }// </editor-fold>//GEN-END:initComponents

    public static JLabel linkify(final String text, String URL, String toolTip) {
        URI temp = null;
        try {
            temp = new URI(URL);
        } catch (Exception e) {
            e.printStackTrace();
        }
        final URI uri = temp;
        final JLabel link = new JLabel();
        link.setText("<HTML><FONT color=\"#000099\">" + text + "<br /></FONT></HTML>");
        if (!toolTip.equals("")) {
            link.setToolTipText(toolTip);
        }
        link.setCursor(new Cursor(Cursor.HAND_CURSOR));
        link.addMouseListener(new MouseListener() {

            public void mouseExited(MouseEvent arg0) {
                link.setText("<HTML><FONT   color=\"#000099\">"   +   text   +   "<br
/></FONT></HTML>");
```

```
            }

            public void mouseEntered(MouseEvent arg0) {
                link.setText("<HTML><FONT  color=\"#000099\"><U>"  +  text  +  "</U><br
/></FONT></HTML>");
            }

            public void mouseClicked(MouseEvent arg0) {
                if (Desktop.isDesktopSupported()) {
                    try {
                        Desktop.getDesktop().browse(uri);
                    } catch (Exception e) {
                        e.printStackTrace();
                    }
                } else {
                    JOptionPane pane = new JOptionPane("Could not open link.");
                    JDialog dialog = pane.createDialog(new JFrame(), "");
                    dialog.setVisible(true);
                }
            }

            public void mousePressed(MouseEvent e) {
            }

            public void mouseReleased(MouseEvent e) {
            }
        });
        return link;
    }

    private  void  jButtonWordActionPerformed(java.awt.event.ActionEvent  evt)  {//GEN-
FIRST:event_jButtonWordActionPerformed

        TextRend = jTextFieldWord.getText().trim();
        String[] keyWords = TextRend.split(" ");
        String[] query1 = new String[keyWords.length];
        String AsImplodedString;

//        String query1 = TextRend.substring(0, ind);
//        String query2 = TextRend.substring(ind + 1, TextRend.length());

        // COMBINED MULTIPLE QUERY WORD

        Vector<String> returnedDocs = new Vector<String>(2, 2);
        Vector<String> returnedDocs2 = new Vector<String>(2, 2);
        Vector<String> FinalRankedDocs = new Vector<String>(2, 2);

        String docsForQuery1[] = new String[keyWords.length];
        String docsForQuery2[] = new String[1];

        // Separater of directory and file name
        Pattern pat = Pattern.compile(",");
        if (TextRend.isEmpty()) {
            JOptionPane.showMessageDialog(null, "Please endet the serach key word!",
"ADIRS: Message", JOptionPane.INFORMATION_MESSAGE);
        } else {
            if (keyWords.length == 1) {

                File fileRendQuery1 = new File("query1.jpg");
                TextToimage Query1Convert = new TextToimage();
                Query1Convert.textRend(keyWords[0], fileRendQuery1);

                File imageQuery1 = fileRendQuery1;
```

```
                String imageNameQuery1 = imageQuery1.getPath();

                PlanarImage piQ1 = JAI.create("fileload", imageNameQuery1);
                SampleModel smQ1 = piQ1.getSampleModel();
                int widthQ1 = piQ1.getWidth();
                int heightQ1 = piQ1.getHeight();

                AllQueryMethods.HorbordQ(piQ1,    hQ,    heightQ1,    hQ,    widthQ1,
pixelaverageQ, smQ1, pwwbQ, hordataStart,
                        hordataEnd, nbands, nbandsQ, pwwvQ, imageNameQuery1);

                returnedDocs  =  AllQueryMethods.MatchingAlgorithms(widthQ1,  heightQ1,
nbands, nbands, widthQ1, heightQ1, nbands, nbands, pwwbQ);

                // Initialize docsForQuery2
                docsForQuery2 = new String[returnedDocs.size()];
                if (returnedDocs.isEmpty() == false) {
                    // Splitting returnedDocs
                    for (int x = 0; x < returnedDocs.size(); x++) {
                        String                  extractedFileName[]                  =
pat.split(returnedDocs.elementAt(x));
                        docsForQuery2[x] = extractedFileName[1].toString();
                    }
                }


                if (returnedDocs.isEmpty()) {
                    System.out.println("Sorry,  there  are  no  documents  in  the  corpus
that contain the query terms you entered.");
                } else {
                    System.out.println("Searching is Finished");

                    // Then, add docs that contain only one of the query terms
                    // Populate it with docs that contain Query2 only
                    for (int j = 0; j < docsForQuery2.length; j++) {
                        if (FinalRankedDocs.contains(docsForQuery2[j]) == false) {
                            FinalRankedDocs.add(docsForQuery2[j]);
                        }
                    }

                    System.out.println("The relevant documents retrieved are:");
                    JFrame frame = new JFrame("Final Search Results");
                    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                    frame.setSize(400, 100);
                    frame.setLocationRelativeTo(null);
                    Container container = frame.getContentPane();
                    container.setLayout(new GridBagLayout());
                    container.add(new JLabel("Click "));
                    for (int k = 0; k < FinalRankedDocs.size(); k++) {
                        int rankOfDocument = k + 1;

                        container.add(linkify(FinalRankedDocs.elementAt(k),
"E:/ado/final/images/" + FinalRankedDocs.elementAt(k), "The results"));
                    }
                    frame.setVisible(true);
                    System.out.println("**********************************");
                } // end of else
            } else {
                /**
                 * Process Query
                 */
                for (int d = 0; d < keyWords.length; d++) {
                    File fileRendQuery1 = new File("query1.jpg");
```

```
                    TextToimage Query1Convert = new TextToimage();
                    Query1Convert.textRend(keyWords[d], fileRendQuery1);

                    File imageQuery1 = fileRendQuery1;
                    String imageNameQuery1 = imageQuery1.getPath();

                    PlanarImage piQ1 = JAI.create("fileload", imageNameQuery1);
                    SampleModel smQ1 = piQ1.getSampleModel();
                    int widthQ1 = piQ1.getWidth();
                    int heightQ1 = piQ1.getHeight();

                    AllQueryMethods.HorbordQ(piQ1,   hQ,   heightQ1,   hQ,   widthQ1,
pixelaverageQ, smQ1, pwwbQ, hordataStart,
                            hordataEnd, nbands, nbandsQ, pwwvQ, imageNameQuery1);

                    returnedDocs2     =     AllQueryMethods.MatchingAlgorithms(widthQ1,
heightQ1, nbands, nbands, widthQ1, heightQ1, nbands, nbands, pwwbQ);

                    // Initialize docsForQuery1
                    docsForQuery1 = new String[returnedDocs2.size()];
                    if (returnedDocs2.isEmpty() == false) {
                        // Splitting returnedDocs
                        for (int x = 0; x < returnedDocs2.size(); x++) {
                            String                extractedFileName[]                =
pat.split(returnedDocs2.elementAt(x));
                            docsForQuery1[x] = extractedFileName[1].toString();
                        }
                    }


                } // end of for loop

                if (returnedDocs2.isEmpty()) {
                    System.out.println("Sorry,  there  are  no  documents  in  the  corpus
that contain the query terms you entered.");
                } else {
                    System.out.println("Searching is Finished");


                    // Then, add docs that contain only one of the query terms
                    // Populate it with docs that contain Query1 only
                    for (int i = 0; i < docsForQuery1.length; i++) {
                        if (FinalRankedDocs.contains(docsForQuery1[i]) == false) {
                            FinalRankedDocs.add(docsForQuery1[i]);
                        }
                    }
                    //  Populate  FinalRankedDocs  with  common  docs  that  contain  all
query terms
                    for (int i = 0; i < docsForQuery1.length; i++) {
                        for (int j = 1; j < docsForQuery1.length; j++) {
                            if (docsForQuery1[i].equals(docsForQuery1[j])) {
                                if    (FinalRankedDocs.contains(docsForQuery1[i])    ==
false) {
                                    FinalRankedDocs.add(docsForQuery1[i]);
                                }
                                break;
                            }
                        }
                    }

                    System.out.println("The relevant documents retrieved are:");
                    JFrame frame = new JFrame("Final Search Results");
                    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
                    frame.setSize(400, 100);
                    frame.setLocationRelativeTo(null);
                    Container container = frame.getContentPane();
                    container.setLayout(new GridBagLayout());
                    container.add(new JLabel("Click "));
                    for (int k = 0; k < FinalRankedDocs.size(); k++) {
                        int rankOfDocument = k + 1;
                        //System.out.println(rankOfDocument    +    "    .    "    +
FinalRankedDocs.elementAt(k));
                        container.add(linkify(FinalRankedDocs.elementAt(k),
"E:/ado/final/images/" + FinalRankedDocs.elementAt(k), "The results"));
                    }
                    frame.setVisible(true);
                    System.out.println("**********************************");
                } // end of else
            }
        }
    } //GEN-LAST:event_jButtonWordActionPerformed
```

## APPENDIX IV: JAVA CODE FOR IMAGE VIEWER AND INTERFACE GUI

### IMAGE VIEWER AND INTERFACE GUI

```
* Author - Biruk Mengistu
 */
package ImageDocuments;

import java.awt.Color;
import java.awt.ComponentOrientation;
import java.awt.Dimension;
import java.awt.Toolkit;
import javax.swing.JTextField;
import javax.swing.*;

public class UI extends javax.swing.JFrame {

    /** Creates new form UI */
    public UI() {
        initComponents();
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
```

```
        jPanel2 = new javax.swing.JPanel();
        jLabel3 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        JTextField jTextField1 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));

        jLabel1.setFont(new java.awt.Font("Microsoft Sans Serif", 1, 18)); //
NOI18N
        jLabel1.setText("Amharic Document Retrival System");

        jLabel4.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ImageDocuments/searching.jpg")
)); // NOI18N

        javax.swing.GroupLayout          jPanel1Layout          =          new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jLabel4)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.R
ELATED)
                .addComponent(jLabel1,   javax.swing.GroupLayout.DEFAULT_SIZE,
369, Short.MAX_VALUE)
                .addContainerGap())
        );
        jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(21, 21, 21)
                .addComponent(jLabel1,  javax.swing.GroupLayout.PREFERRED_SIZE,
37, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addComponent(jLabel4)
        );

        jPanel2.setBorder(javax.swing.BorderFactory.createEtchedBorder());

        jLabel3.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N
        jLabel3.setText("The Search Results :");

        javax.swing.GroupLayout          jPanel2Layout          =          new
javax.swing.GroupLayout(jPanel2);
        jPanel2.setLayout(jPanel2Layout);
        jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addGap(18, 18, 18)
```

```
                .addComponent(jLabel3,  javax.swing.GroupLayout.PREFERRED_SIZE,
134, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(419, Short.MAX_VALUE))
        );
        jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jLabel3,  javax.swing.GroupLayout.PREFERRED_SIZE,
22, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(30, Short.MAX_VALUE))
        );

        jLabel2.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        jLabel2.setText("Insert Search Key Word : ");

        jTextField1.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
        jTextField1.setMaximumSize(new          java.awt.Dimension(2000000000,
2147483647));

        jButton1.setFont(new java.awt.Font("Tahoma", 0, 14)); // NOI18N
        jButton1.setText("Search");

        javax.swing.GroupLayout             layout             =             new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jLabel2)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.U
NRELATED)
                .addComponent(jTextField1,
javax.swing.GroupLayout.DEFAULT_SIZE, 157, Short.MAX_VALUE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.R
ELATED)
                .addComponent(jButton1)
                .addGap(158, 158, 158))
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel2,   javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
            .addComponent(jPanel1,       javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addComponent(jPanel1,  javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
                .addGap(65, 65, 65)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.BASELINE)
                    .addComponent(jTextField1,
javax.swing.GroupLayout.DEFAULT_SIZE, 32, Short.MAX_VALUE)
                    .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE,                                    24,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jButton1,
javax.swing.GroupLayout.PREFERRED_SIZE,                                    34,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(52, 52, 52)
                .addComponent(jPanel2,   javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
        );

        pack();
    }// </editor-fold>

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {
                Dimension                    screenSize                    =
Toolkit.getDefaultToolkit().getScreenSize();

                UI adirsUI = new UI();
                adirsUI.setExtendedState(javax.swing.JFrame.MAXIMIZED_BOTH);

adirsUI.setComponentOrientation(ComponentOrientation.LEFT_TO_RIGHT);
                adirsUI.setSize(screenSize);
                adirsUI.setTitle("Amahric Document Image Retrival System");
//                adirsUI.getContentPane().setBackground(Color.white);
                adirsUI.setVisible(true);
                adirsUI.setResizable(false);

adirsUI.setDefaultCloseOperation(javax.swing.JFrame.EXIT_ON_CLOSE);
            }
        });
    }
    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
}
```

# Appendix V – The Full Amharic script (FIDEL) Set

# Declaration

I, the undersigned, declare that the thesis is my original work and has not been presented for a degree in any other university, and that all source materials used for this thesis have been duly acknowledged.

_____

Biruk Mengistu

Date: 1st June 2014

This thesis has been submitted for examination with my approval as university advisor.

_____

Million Meshesha (Ph.D)

Date: 1st June 2014