# Generalized Haar-like filters for document analysis : application to word spotting and text extraction from comics

Adam Ghorbel

UNIVERSITÉ DE LA ROCHELLE

ÉCOLE DOCTORALE S2IM

Laboratoire Informatique, Image et Interaction (L3i)

Et Laboratoire LIPADE-SIP Université Sorbonne Paris Cité

THÈSE

Présentée par :

Adam GHORBEL

soutenue le 18 juillet 2016
pour l'obtention du grade de Docteur de l'Université de La Rochelle

Discipline : Informatique et applications

# Filtres généralisés de Haar pour l'analyse de documents. Application aux word spotting et extraction de texte dans les bandes dessinées.

---

JURY :

| | |
|---|---|
| **Véronique EGLIN** | Professeur, INSA de Lyon, France, Rapporteur |
| **Laurence LIKFORMAN** | MCF HDR, Telecom Paris Tech, Paris, France, Rapporteur |
| **Josep LlADOS** | Professeur, Université Autonoma Barcelona, Spain. |
| **Jean-Marc OGIER** | Professeur, Université La Rochelle, France, Directeur de thèse |
| **Georges STAMON** | Professeur, Université Sorbonne Paris Cité, France |
| **Nicole VINCENT** | Professeur, Université Sorbonne Paris Cité, France, Directeur de thèse |

**To my parents,**

# Acknowledgments

# CONTENTS

# List of tables

# List of figures

*"Strive not to be a success, but rather to be of value"* **Albert Einstein**

# *List of abbreviations and terminologies*

A list of commonly used abbreviations and terminologies that are used throughout the thesis:

ASCII: American Standard Code for Information Interchange

ADC: Asymmetric Distance Computation

BCC: Basic Connected Component

BLSTM: Bidirectional Long-Short-Term Memory

BoVW: Bags of Visual Words

CC: Connected Component

C-HMM: Continuous Hidden Markov Model

CPDH: Contour Points Distribution Histogram

CWs: Candidate Words

DA: Document Analysis

DAR: Document Analysis and Recognition

DCT: Discrete Cosine Transform

DDTW: Derivative Dynamic Time Warping

DP: Dynamic Programming

DTW: Distance Time Warping

EDM: Euclidean Distance Mapping

EHOG: Elastic Histogram of Oriented Gradient

GHFs: Generalized Haar-like Filters

GMM: Gaussian Mixture Model

GP: Graphical Processing

GW: Georges Washington

HAC: Hierarchical Ascendant Classification

HKS: Heat Kernel Signature

HMM: Hidden Markov Model

HOG: Histogram of Oriented Gradient

II: Integral Image

INT: Index Numeric Table

IR: Information Retrieval

LFS: Local Feature Sequence

LGH: Local Gradient Histogram

LSA: Latent Semantic Analysis

LSI: Latent Semantic Indexing

MCC: Multi-Scale Convexity Concavity

NN: Neural Network

OCR: Optical Character Recognition

PCA: Principal Component Analysis

PHOC: Pyramidal Histogram of Characters

PQ: Product Quantization

QBE: Query-Based-Example

QBS: Query-Based-String

RLSA: Run Length Smoothing Algorithm

RNN: Recurrent Neural Network

SAT: Summed Area Table

SC-HMM: Semi Continuous Hidden Markov Machine

SIFT: Scale-Invariant Feature Transform

SLH: Scott and Longuet Higgins

SOM: Self Organizing Maps

SSD: Sum of Squared Distance

SSHOG: Slit Style Histogram of Oriented Gradient

SVM: Support Vector Machine

SVT: The Street View Text

TP: Textual Processing

VPs: View Points

VLAD: Vectors of Locally Aggregated Descriptors

WS: Word Spotting

WSC: Word Shape Code

ZOI: Zone of Interest

*"I hated every minute of training, but I said, 'Do not quit. Suffer now and live the rest of your life as a champion' "* **Muhammad Ali**

# I. Introduction

## Contents

# 1. INTRODUCTION

For a long time, paper has been the traditional support for storing, recording, transmitting, and exchanging all types of information. In the beginning of the 19th century, the British mathematician, cryptanalyst, and theoretical biologist ALAN TURING (1912-1954) came with defining the concepts of computation and algorithm with the Turing machine. This invention is considered as the first model of a general purpose computer (M. H. A. Newman 1955, 1912–1954). He was then considered as one of the fathers of theoretical artificial intelligence and computer science (P. Gray 1999). Years after years, the world has seen major evolution in the computer science and electronics. This provides technological solutions to a bunch of real problems. One of those problems consists in studying and analyzing the large amount of documents such as books, journals, scientific letters, etc. The ultimate solution was to digitize those handwritten and printed documents so that they would be readable thanks to the different computer devices. This was possible after inventing a drum scanner at the US National Bureau of Standards in 1957. As a consequence, computer scientists create a new scientific field known as Document Analysis and Recognition (DAR).

DAR is considered as the use of both image processing and image analysis dedicated to document images. Image analysis techniques depend on image processing techniques as pre-processing in order to process document images more quickly, accurately, and efficiently. The relationship between these technologies is shown in figure1.

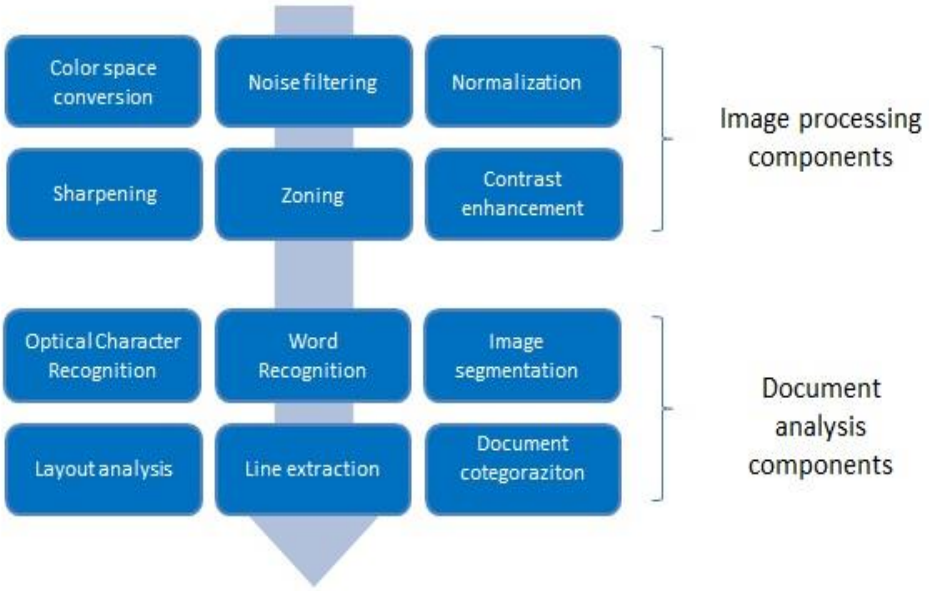


**Figure 1: A set of DAR technologies**

Yet, a DAR system may integrate various fields such as image processing, artificial intelligence, database system, cloud technology, big data, and pattern recognition.

The main objectives of DAR systems are:

- To recognize textual and graphical components in input documents.
- To extract information about different characteristics or parameters of input documents to get the semantics or to extract contents.

In a particular way, document processing could be divided into two subareas which are: Textual Processing (TP) and Graphical Processing (GP). On the one hand, a set of tasks may be derived from TP that are highly related to text components. It goes as: Recognizing the text by Skew determination, text lines and word detection and extraction, the Optical Character Recognition system known as OCR etc. On the other hand, in contrast to TP, GP deals with graphic components. The following diagram (figure 2) shows the different document components dealt with in document processing.

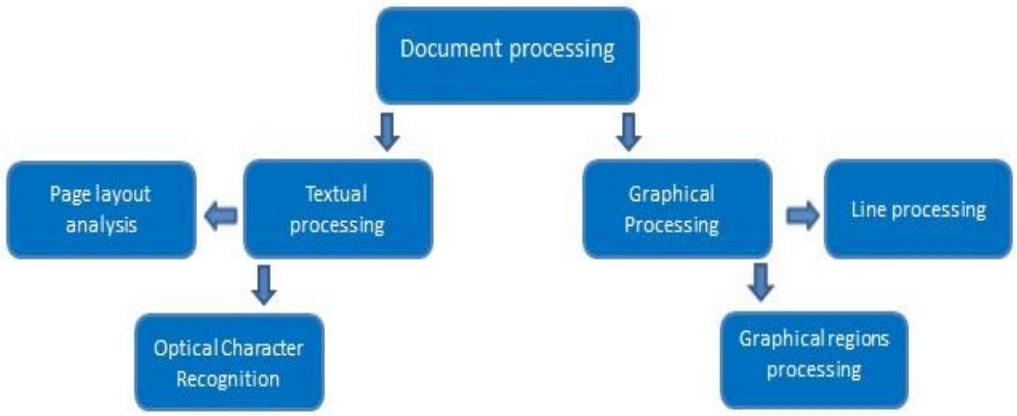


**Figure 2: The document components treated in document processing.**

More globally, DAR systems may integrate supervised and unsupervised classifiers. These classifiers are important to enhance final results in many topics such as: Pixel and region classification, page classification, text recognition, character segmentation, writer identification, script identification, signature verification, and document categorization.

Furthermore, one of the well-known applications of Document Analysis (DA) is Information Retrieval (IR). Indeed, we may find various information retrieval software in the textual field, but they cannot be applied on digitized documents that are increasingly available on the web or in digital libraries. Moreover, information retrieval is usually performed by text analysis methods. Generally, the most efficient processes in text analysis deal with documents that are ASCII files. Actually, the performance of the Optical Character Recognition (OCR) engines is still too poor, especially for manuscript recognition, to transform the document images in ASCII files. Then, OCR does not present a complete solution to the problem because of its limitations in dealing with handwriting and degraded ancient collections. In fact, OCR accuracy is not high enough with handwritings in an open vocabulary context.

Thus, Word Spotting (WS) is considered as an alternative to traditional OCR for different applications such as indexing and retrieval in digitized document collections. We define word spotting as a task that aims to find multiple occurrences of a query in document images. The Query is usually a

word represented by an image or fixed by the user in ASCII form. Word Spotting facilitates indexing and retrieval of information while analyzing the occurrences of specific queries in historical or modern documents. It is very efficient when documents are relatively complex and degraded and are not easily to be retro converted to ASCII text. Yet, conceiving a word spotting framework would extend the scope of RI's software and gives a new research direction to DAR scientists in order to dig more deeply into the RI field.

Before we introduce the specification of the needed features and techniques and design the architecture of such framework, we intend to come up with an inspiring approach that will be simple in use, efficient in computational time, and will open new visions for DAR researchers. So, we based our work on evidences accumulated by both physiological and psychological assumptions. Two capital problems may be highlighted:

- What are the different features and properties that characterize string queries that we are looking for?
- How to combine the generated characteristics together to overcome constraints due to the different adopted techniques in order to obtain final accurate results?

When we focus our attention on an entire document page, some distinction may be attained. At first sight, we discriminate regions with text, regions with graphic components such as logos, tables, pictures, curves, etc. As a first visual approximation, our visual system is able to segment the document into regions. Each region is constituted by different contents sharing the same nature. Then, some assumptions have to be made. For instance, text regions would have rectangular shapes, most of the time horizontal shapes; pictures such as logos or graphical images would have various types of shapes. Thus, if we want to identify a word visually, we have to distinguish the horizontal shapes associated with potential text lines and then we focus more and more deeeply on these shapes to extract a smaller shape that represents the word. Then, our visual system decomposes the document along different dimensions. As a matter of fact, we conclude that this preattentive process is a multi-level one: it gets started from the top (i.e. the whole document) to the bottom (i.e. smaller entities such as words or characters).

Being inspired by this preattentive vision process, we have had the idea of applying some techniques for feature extractions in the word spotting framework at a high level, globally on the entire page image, without any knowledge about text line position. This responds to the fact that our visual system focuses first on the results obtained globally by detecting all text areas then focuses more precisely to detect smaller entities as words.

Furthermore, the general principle of the human perception assumes that first, individual features will affect perception, then these features are gathered together to obtain objects that influence our perception. This inspired us to multiply the number of viewpoints, which permits to focus on relevant parts in the document image.

As a conclusion, the preattentive phase and more generally the human perception processing has inspired us conceiving and developing a coarse-to-fine structure.

# 2. MOTIVATION

National or regional repositories as well as companies' repositories contain a rich heritage of collections of letters, verbal assaults, birth-certificates, marriage-certificates, medical reports, photographs, posters, cards, maps, etc. These documents cover all topics and they are well-preserved since decades. Furthermore, there are tons and tons of manuscript archives that are always indispensable in various administrative operations.

Considering the vast amount of information within those existing documents, huge human efforts are required to explore and manipulate the overall contents. Such activities need considerable amount of time to be performed. So, as a matter of fact, administration employees confront such constraints every time they want to retrieve textual information such as words or signatures and graphical information like logos or tables.

One of the aims of the different projects of the French administration is to speed up the process of retrieving textual information especially the research of words within the existing documents so they will be able to analyze their archives efficiently, properly, and accurately. So, administration officials would like to develop a digitized policy in order to automatize the process of archives analysis. This provides a valorization of these documents through numerical usage and virtual exhibition.

Yet, a computer based framework enabling searching and retrieving textual information is recommended to respond to such constraints. This is encouraged by the valuable advantages given by micro-computer and electronic technologies. One of the most major and important tasks in this domain is the process of searching and indexing a sequence of characters, characterizing words or texts, through documents.

Thus, the objective in this thesis is to design and to develop a robust word spotting framework dedicated to manuscript documents. More specifically, the framework must automatically retrieve a word, which is introduced in ASCII form by the user, from manuscript document databases. The framework will be then in future work implemented to practical systems for word retrieval in the administration document databases or in virtual libraries. It facilitates the search for textual contents with refined and accurate mechanism and functionality. Furthermore, such framework is a primordial computer assistant for users in order to explore digitized manuscript or even printed resources as letters, mails, books, etc. Moreover, it helps classify the various documents by searching keywords that characterize the type of the document found with it such as bills or birth-certificates. Additionally, it is a well needed framework by many companies in order to automatize the classification of documents, to obtain automatic rankings for archives and to automatize the document analysis process.

# 3. OBJECTIVES AND CONTRIBUTIONS

In this section, we are going to present the various objectives and contributions of the thesis.

## 3.1. Objectives

The fundamental purpose of the thesis is to facilitate the work of organizations, companies, and researchers that study and explore digitized handwritten or printed modern or historical documents

which are relatively complex, by providing advanced information retrieval and indexing functionalities by word spotting technology.

The main objective is to introduce a word spotting framework for manuscript documents where the Optical Character Recognition systems (OCR) cannot provide accurate textual information. Thus, major scientific difficulties that must be overwhelmed in this thesis are linked to handwritten document analysis.

One of the various objectives of this work is to enable the user to express the query in an ASCII form thanks to a keyboard input. Indeed, this enables the research of queries even if they are not present within processed documents. However, without any hypothesis on the document characteristics such as size of characters and fonts, real problems are going to occur.

Furthermore, a specific aspect that we would like to avoid is a training stage on a database; this requires designing a system capable of adaptation and self-learning. Furthermore, as such writing characteristics in manipulated handwritten documents are not *a priori* known, the proposed system must be able to automatically detect some of these characteristics such as the average size of writing characters for instance.

Moreover, we would like to introduce a word spotting approach that does not rely on any a priori fixed set of characteristics, but is based on a family of features that will simultaneously fit the search term and the document within the performed process.

Another objective of this thesis is the manipulation of different types of documents presenting a wide variability at different levels. Other than these several interesting points, we would like that the word spotting framework will be analytical, meaning that it will not be based on document layout segmentation. Besides, no binarization phase is recommended in order not to lose various crucial information.

Apart from all these objectives concerning the word spotting task, and to illustrate how the method we propose can handle different observation levels as well as different purposes, we have chosen to apply globally the adopted features in the word spotting phase on complex graphical documents as comics. Here, the objective would be first the detection and localization of texts within pages from comics in order to separate it from the graphical contents. Thus, we propose an original approach that will not require any *a priori* detection of balloons or panels or other graphical contents in order to detect and localize textual information.

## 3.2. Contributions

While studying and exploring the conjectures mentioned above, various contributions have been brought out by the end of this thesis. We are going to recapitulate them in this section.

### 3.2.1. Bio-inspired approach

The proposed framework is essentially based on some human perception characteristics. It was inspired by two characteristics of human vision:

- Preattentive processing that leads to develop a coarse-to-fine structure.
- The principle of the human perception which assumes that first the different individual features influence perception then these features are gathered together to obtain object recognition.

These properties help us to introduce a multi-scale approach that is propagating from coarse to fine scales and to multiply the number of viewpoints allowing focusing on relevant parts in processed documents. They allow introducing an approach that is based on two modules performed sequentially at two different observation scales:

- A global filtering permits reducing the search space into zone of interests containing candidate words.
- A refining phase permits retaining only candidate words which are the most similar to the query.

Furthermore, our coarse-to-fine approach is an analytical one that is applied on the entire processed document images without any use of segmentation techniques.

### 3.2.2. Scientific research contribution

A comprehensive survey of the past researches in word spotting based techniques in both handwritten and typewritten document images has been done. We have proposed a classification of word spotting approaches that have been introduced in the literature since 30 years from now. Till now, this original research may be considered as reference of word spotting tasks for all researches. It highlights:

- A comprehensive study of word spotting approaches
- A new classification of the word-spotting approaches according to different criteria
- A clear and guided overview for readers about word spotting approaches
- A relationship between word spotting approaches through their advantages, disadvantages, and evaluation results.

### 3.2.3. Simplicity and efficiency

The use of Haar-like features that are easily computed from the integral image of each processed document image. These filters are so simple to be used and possess valuable properties as follows:

- Evaluated at different observation scales.
- Computed at distinct orientations.
- Not memory consuming.
- Sensitive to the presence of edges and other simple image structure.

Over all other features used in image analysis, Haar-like features may be considered as one of the simplest ones. They are able to extract various features and to solve a large variety of problems.

Besides, the selection of the right Haar-like filters and the right scale has to be achieved according to the word and to the characteristics of processed document images. Thus, the simplicity of these filters allows us to generalize and adapt them to various characteristics of words within manipulated document in order to obtain accurate final spotting results.

### 3.2.4. Originality

In this thesis we have proposed two approaches for word spotting and text localization in handwritten documents and comics respectively. The main contribution of these two approaches is the ability to

manipulate different types of documents especially handwritten collections that present a wide variability at different levels and also graphical documents that contain various textual and graphical contents such as comics.

- Handwritten documents: they may have fragmented characters, variability in writing style, overlap of components like components belonging to several text lines because of the presence of ascenders and descenders, crossed out words, overlapped writings, noisy background, etc.
- Comics: they are considered as complex graphical documents. Their background is of a graphical nature. The textual elements can be randomly situated and oriented with various fonts, style, alignment, size, and colors. Indeed, there are various content types within each comic document.

Furthermore, original techniques and studies have been exposed in the two research sections we explored in this thesis:

- **Word Spotting in handwritten documents**

Our word spotting approach is based on some delightful assumptions:

- No layout segmentation of the processed documents: there is no segmentation and the query is searched within the document.
- No binarization step of the processed documents to avoid losing data preventing from discriminating roughly similar words.
- No learning step in the matching process which let us to avoid the process of extraction features a priori and the learning phase.

Besides, it is based on original techniques such as:

- Haar-like features are used for the first time in word spotting task: As far as we know and after performing a whole comprehensive survey for word spotting in the literature, we remark that Haar-like features are never used through the process of word spotting in both handwritten and printed documents. Yet, we open a new research path for Scientifics in order to further explore this problem.
- New technique for modeling input string queries: Aside from different techniques that create synthetically the query with specific fonts and sizes, we propose to represent each query with a prototype composed by adjacent rectangles.
- A new technique is described which permits estimating the average size of handwritten characters written in the processed manuscript.
- An accumulation process of votes: this process allows, by translating the different votes which indicates the presence of a specific pattern in the neighborhood of a pixel, obtaining zones of interest (ZOIs). Consequently, these zones of interest correspond to the possible presence of the query in this spatial region.
- An automatic technique for obtaining a threshold value, which permits visualizing the ZOIs within the document image.

- **Text and graphic separation in comics**

The text and graphic separation approach is based on some interesting assumptions:

- ❖ It does not rely on assumptions about the text localization, the written color, and the style of the text.
- ❖ It is done globally on the processed comic page image without applying any layout (i.e. frame or speech balloons) detection techniques.

Moreover, this approach is also based on:

- The use of Haar-like features that are applied for the first time in text detection and localization in comics: This results in detecting text areas in a constant complexity independently of the different graphical contents within the processed comic page images.

# 4. OUTLINE

The thesis is organized in 5 other chapters as follows:

**Chapter II:** presents a comprehensive survey of the past researches in word spotting. This chapter describes the proposed word spotting approaches in the literature and summarizes the most important features used in different word spotting approaches and also enumerates the different methods that have been used in the spotting process. Finally, it gives various advices and recommendations to the community of word spotting and information retrieval.

**Chapter III:** reviews for Haar function and Haar-like features. It describes the process of transforming a grey scale image to an integral image representation. Moreover, this chapter outlines the various applications using Haar like filters.

**Chapter IV:** is dedicated to the challenge of separating text from graphics in comics. At the beginning, a survey of text detection and localization of text in comics is established. Then, the text detection phase and text/graphic separation process are detailed. At the end, quantitative and qualitative evaluations on the eBDthèque database of comics are performed.

**Chapter V:** introduces the bio-inspired coarse-to-fine word spotting approach. The process of modeling the string query typed by the user is presented. Then, the principle two phases of the proposed approach are detailed. In the global phase, the generalizing process of Haar-like filters on the processed documents process is described. Then, the process of estimating the different parameters allowing the adaptation to the document is mentioned. After that, the vote accumulation technique is introduced. After this phase, the process of decision making and generating the final results is outlined. In the refining phase, the process of improving the obtained results from the previous phase is described. Finally, quantitative and qualitative evaluations on George Washington database and Bentham database are performed.

**Chapter VI:** concludes the thesis and defines the future of word spotting and comic document analysis.

*"To raise new questions, new possibilities, to regard old problems from new angle, requires creative imagination and marks real advance in science"* **Albert Einstein**

# II. State of the art

## Contents

# 1. ABSTRACT

Word spotting techniques allow searching multiple occurrences of an input query by using matching methods in document images. It also allows creating partial indexes for the processed document collections similar to the back-of-the-book indexes. Thus, word spotting facilitates the indexing and retrieval of information given as a query in digitized historical and modern documents relatively complex and degraded. In this chapter, we present a comprehensive survey of the past researches in word spotting based techniques in both handwritten and typewritten document images.

# 2. INTRODUCTION

In documents, information retrieval is usually performed by text analysis methods. The performance of the Optical Character Recognition (OCR) engines is still poor, especially for manuscript recognition. OCR does not present a complete solution to the problem because of its limitations in dealing with handwriting and degraded ancient collections. In fact, OCR techniques cannot be achieved accurately because character recognition systems are not well suited for handwritings in an open vocabulary context. For that, word spotting is considered as an alternative to traditional OCR for different applications such as indexing digitized document collections and retrieving information.

In the literature, word spotting approaches have been applied to various scripts such as Latin, Arabic, Greek, etc. These scripts are not the same and they differ by alphabet sizes, writing direction, cursiveness and similarity among characters. They can be either handwritten or typewritten. Moreover, word spotting approaches have been divided into different categories in multiple ways by document analysis researchers. For instance, they can be divided into two main categories based on matching techniques, image based matching techniques and feature based matching techniques (J.L Rothfeder, S. Feng, T.M. Rath. 2003). The former include methods that compute word distances directly on image pixels such as template matching using correlation while the latter compute divers feature in word images and then those features are matched. Another classification can be found in (J. Lladós, M. Rusiñol, A. Fornés, D. Fernández and A. Dutta 2012). Two main approaches of word spotting exist depending on the representation of the query. They can be based on Query-by-string (QBS) or on Query-by-example (QBE).

The QBS methods (H. Cao and V. Govindaraju. 2007) use character sequences as input. It typically requires a large amount of training materials since characters are learned a priori and the model for a query is built at runtime from the models of its constitutive characters.

In QBE methods (R. Manmatha, C. Han ,E. M. Riseman 1996), the input is one or several exemplary images of the queried word. Therefore, these methods require collecting one or several examples of the queried word. Another popular categorization technique divides the methods into either segmentation based methods or segmentation-free methods as in (T. Adamek, N.E. O'Connor, A.F. Smeaton 2007) (B. Gatos and I. Pratikakis. 2009). We, then, classify the different word spotting approaches into two main categories which are: Holistic analysis techniques and Analytical recognition techniques. Furthermore, we divide each main category into: QBE based techniques and QBS based techniques.

Holistic word based techniques are basically segmentation-free techniques. A word image is a unit no further segmented. At the opposite, analytical techniques are segmentation-based techniques in which a document image or word image is segmented into smaller units which can be recognized independently or when grouped.

Most of the techniques in literature are belonging to the holistic analysis class because they have been applied on handwritten documents where characters segmentation is very difficult to be done in an accurate way. However, recently, some analytical analysis based techniques have been proposed as well, with the objective of achieving accurate recognition rates.

In this chapter, we focus on detailing most of word spotting based approaches in the literature and give some of their results without comparing them because there is no standard criterion and there is also a lack of standard databases on which to compare the evaluations. In fact, these techniques have been experimented on several types of data sets with different number of tests and queries. All of that makes the common measures for criticizing the quality of the obtained results do not accurately describe the efficiency of a particular technique with respect to others. Besides, the aims of the different applications may differ. As far as we know, there are some researches that have been trying to find an evaluation framework as (K. Terasawa,H. Imura, Y. Tanaka 2009) who proposed an automatic evaluation framework which helps to obtain a uniform standard in the evaluation process, by providing certain protocols and guidelines. Recently, a framework evaluating segmentation-free word spotting approaches without Hard Decisions is proposed in (W. Pantke, V. Margner, T. Fingscheidt 2013). This work is a complement to the evaluation process for the symbol spotting task presented in (M. Rusinõl , J. Llados 2009).

In the rest of this chapter, the main word spotting approaches in the literature are summarized. The classification of word spotting approaches is presented in section 3 and section 4. In section 3, the different QBS and QBE based word spotting holistic techniques are described. A review of QBS and QBE based word spotting analytical techniques is given in section 4. Finally, the section 5 and the section 6 concludes this chapter by bring out several conclusions and remarks.

# 3. HOLISTIC ANALYSIS TECHNIQUES

Holistic word recognition techniques consider word images as units. They mostly rely on a word segmentation process. In fact, the final obtained spotting results rely upon the quality of the segmented words in the document images. So, for the sake of completeness and in order not to combine two different problem levels, we give an overview of some word segmentation techniques used in the pre-processing stage of holistic word spotting systems.

The next figure (figure 3) shows an overview of the different holistic techniques.

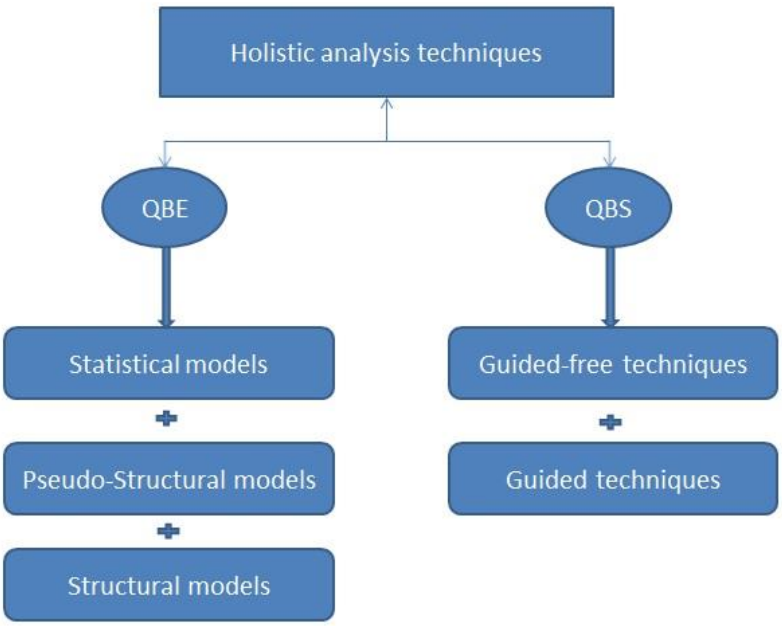


**Figure 3: Flowchart of the holistic techniques.**

### 3.1.1. Word segmentation techniques

Several works in word segmentation including many approaches for machine printed and handwritten documents are cited in (M. Makridis,N.Nikolaou,B.Gatos 2007). They rely on either blob extraction or on a learning process based on the size of the intervals between two consecutive words.

A scale-space word segmentation process is proposed in (R. Manmatha, J. Rothfeder 2005). This system is performed on the George Washington manuscripts dataset. In this work, the page margins are removed. Then a gray-level projection profile algorithm extracts lines in the images. Each line is then filtered with an anisotropic Laplacian at several scales. Blobs are produced. They correspond to portions of characters at small scales and to words at larger scales. Scale selection is done by finding the maximum over scale of the area of blobs. The blobs recovered at the optimum scale are then bounded with rectangular boxes to recover the words. Finally, boxes of unusual size, unlikely to correspond to words are eliminated.

In (S. Srihari,H.Srinivasan, P. Babu, C. Bhole 2005), a neural network word segmenting algorithm is presented. This algorithm is applied on Arabic manuscript where words are well separated with regular spaces. Neural network is used for deciding whether or not the space is between two distinct words. Besides, a word segmentation technique for historical and degraded documents is described in (M. Makridis,N.Nikolaou,B.Gatos 2007) but dedicated to printed documents. This technique is based on three steps. The first one is based on a dynamic run length smoothing algorithm that helps grouping together homogenous text regions. The second one is based on noise and punctuation marks removal in order to facilitate the segmentation process. The final step consists on a draft text line estimation procedure that guides the final word segmentation. All these steps make this technique deal with texts of different sizes within a document, text and non-text areas lying very near and with no-straight and skewed text lines. Recently, two novel approaches based on a learning mod-

el to extract text lines and words from handwritten documents are presented in (V. Papavassiliou, T. Stafylakis, V. Katsouros, and G. Carayannis 2010) where the authors used a SVM-based metric to locate words in each text line segmented by a Viterbi based technique.

After describing the most important used word segmentation techniques, we are going to give now an overview of the word spotting holistic based approaches.

### 3.1.2. Word segmentation based techniques

In this section, we divide the holistic word based techniques into two sub-classes that entirely depend on the query formulation which can be Query-by-example or Query-by-string techniques.

#### 3.1.2.1. Query-by-example (QBE) based approaches

Word spotting for indexing document images and for information retrieval was first used in the literature in (R. Manmatha, C. Han ,E. M. Riseman 1996) and (R. Manmatha, C. Han, E.M. Riseman, W.B. Croft. 1996). Later on, (T. Rath, R. Manmatha 2007) proposed a word spotting approach for document image retrieval as well as for indexing historical document images. This technique allows grouping together word images into clusters of similar words. These approaches take as input one or several sample images of the queried word, then, they are known as Query-by-Example (QBE) based approaches. The accuracy of the QBE based methods relies upon the word segmentation and representation steps. Each segmented word is represented by features depending on the type of processed documents. In most of the holistic word spotting approaches, each word is presented by one of a three types of models that can be classified into statistical, pseudo-structural and structural descriptors.

Next, we will mention some holistic word spotting approaches based on these models.

##### 3.1.2.1.1. Statistical models

The statistical descriptors represent the image as an n-dimensional feature vector. They can be defined from global and local features. For the former, scalar features, that are computed from the whole image, such as height, width, aspect ratio can be used. However, local features are computed from local regions from the image or even from primitive extracted from them. For instance, these features can be dots or crosses at key-points or region, position/number of holes. Generally, local features and global texture features provide different information about the image because the support over which texture is computed varies. However, combining both of these features is beneficial as rough segmentation of objects are available (D.A. Lisin, M.A. Mattar, M.B. Blaschko, M.C. Benfield,E.G. Learned-miller 2005). We can say the local features are typical in representing word images but they are not reliable. Contrariwise, the global features have good reliability but they are not typical in representing word images.

A statistical word spotting based method is proposed by (R. Manmatha, W.B. Croft 1997). Some statistical features as areas and aspect ratios are used to describe word images. After word extraction, the authors introduced two matching algorithms to rank matches of words with a given query image. The first matching technique is based on the Euclidean Distance Mapping (EDM). However, the second one is based on an algorithm of Scott and Longuet Higgins (SLH). These two techniques are tested on two handwritten pages. The first document contains 192 words and the second one contains 153 words written by a second writer. Experiments show that EDM performs poorly in the case of bad handwritten style due to the fact that it does not resolve the problem of distortions. But, the SLH performs well in the same case. Different features were already evaluated in (T.M. Rath, R. Manmatha 2003). For the representation step, single-valued features such as Projection profile, Upper/Lower word profile, Background to Ink transition, and grayscale variance are tested. In the matching step, the Dynamic Time Warping (DTW) with Euclidean distance enables to match the feature sequences of two words. Experiments are carried out using 2381 words taken from the George Washington manuscript database. The authors used 15 queries to compute the performance of each individual type of feature. Obtained results show that the upper word profile feature performs the best with an average precision of 64%.

Similar work using information about profiles is described by (A. Kołcz, J. Alspector, M. F. Augusteijn, R. Carlson, and G. V. Popescu 2000). The authors proposed an approach to word spotting in handwritten documents based on the principles of shape similarity. They applied a DTW algorithm for matching the word images. Three profile features are found out for each word, the top-most transition position, the bottom-most transition position and the number of ink-background transitions. They are resistant to noise and can be interpreted as crude measures of word-image complexity. Indeed, the use of these holistic features makes this proposed method effective even if the lexical structure of the handwriting is ambiguous. For experiments, they selected 19 examples of images for four common words. These examples, known also as models, are taken from a data set containing 13 handwritten documents belonging to the archive of the Indies. The word image comparison algorithm is matching the provided templates to segmented manuscript lines. The obtained results show that this technique works better for longer words as compared to shorter words. This proposed method is highly related to the availability of the models in the existing data sets which is considered as a limitation for it.

A statistical description method based on the Harris corner detector for each segmented word is used in (J.L Rothfeder, S. Feng, T.M. Rath. 2003). Relative corner correspondences are computed using Sum of Squared Distances (SSD) error measures for comparing the gray level intensity windows centered on the detected corner points. In fact, the correspondences between pairs of feature points capture the similarity between local regions. An example of detected corners and recovered correspondences in two word images is shown in Figure 4.

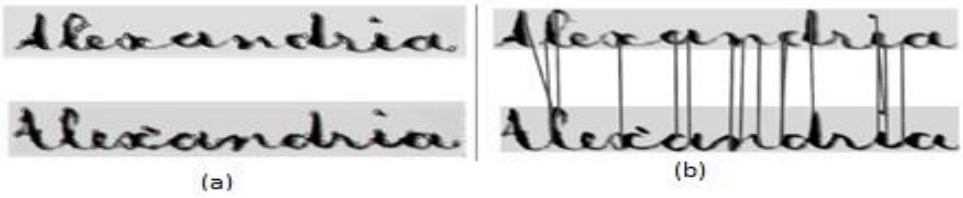


**Figure 4: (a) Corners detected with the Harris Corner detector on two gray level images. (b) Recovered correspondences in two word images (J.L Rothfeder, S. Feng, T.M. Rath. 2003).**

Two words to be matched need to be of the same size, this is why all candidate words are first resized to the size of the query word. Finally, the Harris detector technique is used essentially because of its repeatability, invariance to viewpoint changes and invariance to illumination changes properties. For experiments, the authors used two different quality sets. One contains 2372 images of reasonable quality and the second one contains 3262 images of poor quality. The average precision rate obtained is respectively 62.75% and 15.49%.

The work of (J. Li, Z.G. Fan, Y. Wu ,N. Le. 2009) is based on using a Local Feature Sequence (LFS) to describe the whole image. This sequence is obtained by computing each word length expressed in pixels in each word. The local feature representation is robust and efficient because the word length is not sensitive to image quality. Thus, it is considered as a stable feature. A query is a portion of a document image and can be either a text line or a paragraph; it is given as the input of the system. The document images containing that query are retrieved by matching feature sequences using a coarse-to-fine strategy that combines the advantages of the two substrings matching algorithms which are Dynamic Programming (DP) and Suffix Tree. In fact, they use the Suffix tree algorithm to perform rough matching and, to further compare the retrieved candidate images, they use the DP algorithm. The proposed technique is tested on different contemporary Latin document images. The obtained results show that this method handles some challenges including low-resolution, different languages, distortion, and rotation. Finally, this method can be extended to be performed on Asian document images by extracting more local features.

A statistical model based on Bag of Visual Words (BoW) is used in several QBE based works as in (R. Shekhar, C.V. Jawahar. 2012) and (I.Z. Yalniz, R. Manmatha 2012). The Scale-invariant feature transform (SIFT) features computed at interest points around key-points are the most popular features for building the (BoW) representation model. It allows retrieving relevant documents from datasets consists of Million documents (R. Shekhar, C.V. wahar. 2012). An overview of the BoW representation is given in Figure5.

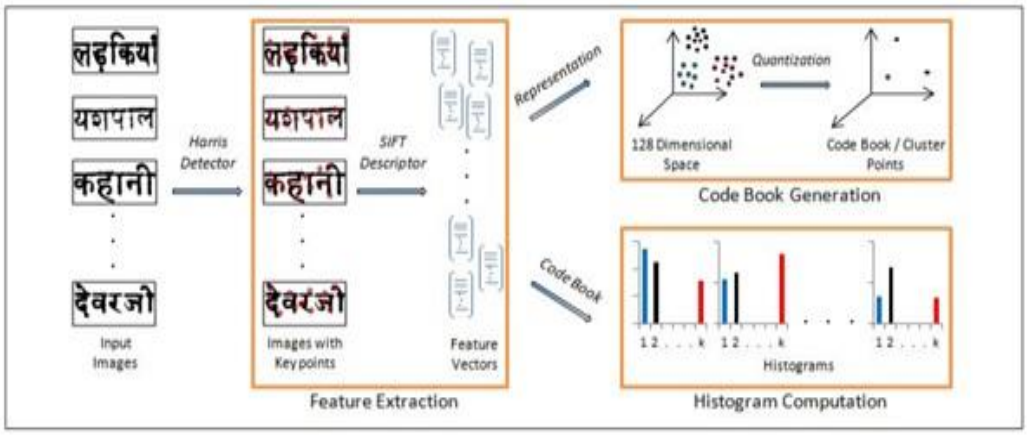


**Figure 5: A Bag of visual Words representation (R. Shekhar, C.V. Jawahar. 2012)**

Indeed, the method proposed by (I.Z. Yalniz, R. Manmatha 2012) is applied on printed and handwritten documents, such as the Indian Telugu books (Telugu-1716 and Telugu-1718) and English book of Arthur Conan Doyle “Adventures of Sherlock Holmes”. The use of the BoW model is robust to occlusions and image deformation. It is also invariant to illumination changes or noise. However, the BoW model does not take into account the spatial relations among the different boxes.

Recently, another BoW based approach is introduced by (M. Rusinõl , J. Llados 2014). This work is proposed to deal with the historical handwritten documents and it takes into account the user feedback. The user, in this framework, is able to provide relevance assessments on the retrieval results. Word shape coding technique using statistical features for word image matching has been used in many word spotting and retrieval approaches in the literature such as in (S. Lu , L. Linlin , C.L.Tan 2008) (S. Bai, L.L. Tan 2009) (T. Adamek, N.E. O'Connor, A.F. Smeaton 2007).

In this context, (T. Adamek, N.E. O'Connor, A.F. Smeaton 2007) proposed a Query-by-example approach for historical handwritten manuscripts based on matching word contours using DTW instead of whole images or word profiles. This work is inspired from (T. Rath, R. Manmatha 2003) (V. Lavrenko, T. Rath, R. Manmatha 2004) (T. Rath, R. Manmatha 2003). The bounding boxes of the words in the document images are supposed to be known. Separated parts in the words are linked together using position of words base line and height. Then, connected components are labeled; successive components in the word are connected by manually adding a synthetic ink line in the binary image. Then, a contour tracing procedure is done in order to extract a single order contour for each word. To describe the word contours, a Multi-Scale convexity concavity (MCC) is used and gives information about the amount of convexities and concavities at different scale levels. A 1D discrete cosine transform (DCT) is applied to each multi-scale contour point to get a MCC-DCT representation. This extraction step deals with the poor quality typical of manuscripts, such as noise and stained paper, contrast variations such as faded ink and differences in pressure on the writing instrument and finally with the disconnected letters. Indeed, the multi-scale contour based descriptors can effectively capture intrinsic word features by avoiding any segmentation of words into smaller sub-units. In the matching step, a DTW is used to find the alignment along the two contours. Experiments are done on a set of 20 pages from the George Washington collection which contains around 4856 word occurrences of 1187 unique words. The obtained results show an accuracy rate of 83%. They also show that this technique is robust to scaling, rotation, transformation, occlusion and even symmetric transformation.

In (S. Bai, L.L. Tan 2009), for each word bounding box, a total of 7 different features are computed and the word is converted into a Word Shape Code (WSC). The features used are: Character ascenders/descenders, deep eastward and westward concavities, holes, i-dot connectors and horizontal-line intersection. They are extracted by applying a run-length smoothing algorithm. After the extraction process, a sequence of feature codes is associated with each word based on the position and the type of the features. Some globalized sequence alignment similarity measures are used for partial or whole word matching. The authors performed two experiments on different sets of document with different qualities. By randomly choosing 40 words of length between 4 to 12 characters, both partial and whole word spotting are performed. Experiments show that this proposed framework is tolerant to serifs, font styles and certain degree of touching, broken or overlapping characters.

As we saw earlier along the described methods, different matching algorithms can be used such as DTW, Euclidean distance, etc. Moreover, a customized Hausdorff distance for matching two words in image space is employed in (A. Andreev and N, Kirov 2009). The evaluation is done on the printed Bulgarian Chrestomathy text from 1884 that has bad quality, many pages have tilted rows, there are significant variations in gray levels, etc. Concerning the Slavonic handwritten text from 1574, the segmentation is quite good due to the clerkly hand of the writer.

Recently, a segmentation based historical handwritten word spotting is described in (K. Zagoris, I. Pratikakis, B. Gatos, 2014.). For word description, key-points are first detected then gradient orientation features are computed. Every feature for local key-points is calculated upon the quantized gradient angles. This approach is evaluated on two handwritten datasets, Bentham Dataset (D. G. Long 1981) and GW dataset. A set of 1570 GW query words and 3668 Bentham query words are used in the experiments. Those words are appearing in different sizes and frequency. The experiments show

that the proposed Document-Specific Local Features achieve higher rates in terms of Mean Average Precision and Precision than those obtained by both profile-based strategies and SIFT local features.

#### 3.1.2.1.2. Pseudo-Structural models

Some word spotting based approaches accumulate pseudo-structural information in the descriptors in order to represent the word images.

In this context, a pseudo-structural descriptor based on Loci features is described in (D. Fernandez, J. Lladós, A. Fornés. 2011) who proposed a handwritten word spotting approach in old manuscript document images of the marriage licenses of the Barcelona's Cathedral. For word images representation, they have used a pseudo-structural descriptor based on Loci features proposed by (H. A. Glucksman 1967) for the classification of mixed-font alphabets. In fact, a characteristic Loci feature consists of the number of the intersections in four directions which are up, down, right and left. The number of intersections (a black/white transition between two consecutive pixels) is counted for each background pixel in a binary image and each direction. Thus, each key-point generates a code known as Locu number of length 4. For that, the skeleton of the image is computed by an iterative thinning. After organizing the feature code-words in a look up table, the matching process between the vector feature of the query word and all the words of the database is done by using Euclidean and Cosine distances. In the experimentation phase, the authors used 10 keyword queries and tested their technique on a set of 30 historical document images taken from the Cathedral of Barcelona archives. Experiments show that this method generates good results due to the nature of the used descriptor which is able to capture more global information including the structure of the word strokes.

Recently, (D. Fernãndez, P. Riba, A. Fornés, J. Lladõs 2014) computed pseudo-structural and structural features at the feature extraction phase. For the pseudo-structural features, the authors used the Loci descriptor. For the structural descriptor, they adopted a Shape Context descriptor that describes a coarse distribution of the neighboring shape of the generated key points.

#### 3.1.2.1.3. Structural models

In the Literature, the use of structural approaches is less common when dealing with spotting questions. These works describe the image as a set of geometric and topological primitives and generate the relationships between them.

The approach described in (P. Keaton, H. Greenspan, R. Goodman 1997), which is similar to that presented in (R. Manmatha, C. Han ,E. M. Riseman 1996), is based on the extraction of information about concavities. The difference between these two references mentioned earlier is that Keaton et al. did not limit themselves to documents written by a single author; instead they attempted keyword matching in a multi-authored domain exhibiting high variability within each keyword class. To characterize a word, they utilized a combination of global and local features in the form of profile signatures and morphological cavities. A DCT is used to encode some profile features while graph-based models are used to encode the cavity features. In the matching process, first the DCT encoded profile signatures are matched by using a Minkowski distance. Then, the profile matching score is incorporated into the keyword signature graph as an additional feature. Finally, a probabilistic graph matching based on Bayesian evidential reasoning is used in order to find the best match between the keyword signatures and those generated by words lying in the candidate regions. This work is considered as one of the first keyword spotting techniques that dealt with unconstrained and cursive handwritten

forms. Finally, the experiments in this work are done on nearly 100 document pages of manuscripts contained of the Indies Seville in Spain.

Moreover, (M. Rusinõl , J. Llados. 2008) presents the spatial organization of the computed local descriptors by a graph. The key-points are the Harris corners, they are linked to their nearest key-points by an edge in the graph. The authors tested three descriptors; SIFT features, Shape contexts, Hu's Geometric moment invariants and a set of steerable filters. Besides using the spatial organization scheme, they used an indexing structure that is a Hash table to efficiently and quickly retrieve key-points with similar descriptors to the query ones. The obtained results do not permit to evaluate the robustness of this method since only 3 symbols and 3 words are queried. Indeed, the different conducted tests on a small database are affected by the local descriptor and the quantization performed by the hash function.

Recently, (P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Liadõs, A. Fornés 2014a) introduced a coarse-to-fine graph matching. For the feature extraction step, the authors extract three types of structural points to build graphs. Then, Shape Context labelled graphs are used to generate the different attributes of the graph vertices. In the matching step, the authors used a bag-of-small-graphs technique to find the candidate words. Then, a graph edit distance based on DTW alignment algorithm is applied to generate the true positives. Experiments are done on a handwritten Dataset containing 50 scanned marriage licenses of the Barcelona Cathedral written by the same writer (J. Almazãn, D. Fernãndez, A. Fornés, J. Lladõs, E. Valveny 2012). The authors used the same query classes and evaluation protocols used in (J. Almazãn, D. Fernãndez, A. Fornés, J. Lladõs, E. Valveny 2012). This latter describes a word spotting approach based on a coarse-to-fine approach combining two steps: (1) An efficient indexing method to generate all candidate zones and (2) A discriminative appearance model to obtain the true positives.

#### 3.1.2.2. Query-by-string (QBS) based approaches

In the literature, the holistic query-by-string based approaches can be either guided or not which means that sometimes they require a user feedback to improve the spotting results. Thus, we classify these approaches into two categories: QBS guided approaches and QBS guided-free approaches.

Furthermore, there are several techniques to generate the query in the QBS based approaches. Some of these techniques produce word images as a query for the system and the spotting process is done by matching the different characteristics of the image query and the document images, so these methods may be the same as in the different QBE based methods. Henceforth, such methods are mentioned in next sub-sections.

##### 3.1.2.2.1. QBS guided-free approaches

Some holistic word retrieval techniques use Latex to produce the query word (S. Marinai, E. Marino, G. Soda 2003) (S. Marinai, E. Marino, G. Soda 2006). (S. Marinai., S. Fain, E. Marino,G. Soda 2006) presented a general system for performing word image retrieval using Self Organizing Maps (SOM) based on word image clustering combined with Principal Component Analysis (PCA). In this system, the query is a text, an ASCII word, and then a word image is produced by Latex. The query formulation is imaged in the same font as in the document to be processed and then corrupted with a synthetic noise in order to simulate the degradation found on the document. In the image word segmentation process, words are extracted using Run Length Smoothing Algorithm (RLSA) and are divided into six index partitions based on their aspect ratio. A vectorial representation of the words is

obtained; the items contain the average gray level of the pixels in grid cells. Clustering is performed on each subset of the partition. The clustering step of the word images is based on SOM. The idea of a clustering technique is extended from (S. Marinai, E. Marino, G. Soda 2003) that exploited the SOM for character-like object clustering. One advantage of the SOM for word clustering is the spatial organization of the feature map that is achieved after the learning process. Finally, the query words are searched for in the top three clusters which are then analyzed using PCA space to get the final top 20 ranked words. In this work, there is no need for a direct comparison of the query word with each indexed word because the combination of the previous methods allows retrieving efficiently the matched words from large document collections. An extension of this work is made by (S. Marinai, E. Marino, G. Soda 2007a) (S. Marinai, E. Marino, G. Soda 2007b) that consisted in building a framework for document retrieval in digital libraries. This work is also based on some previous work as in (S. Marinai, E. Marino, G. Soda 2006) (S. Marinai., E. Marino,F. Cesarini, G. Soda. 2004) (S. Marinai, E. Marino, G. Soda 2005).

Another holistic word QBS based system is proposed by (A. Balasubramanian , Balasubramanian Million Meshesha , C. V. Jawahar 2006). The authors process a large collection of printed document images by matching image features at word level. For representations of the word, profile based and shape based features are employed. Some of these features provide the sequence information while others capture the structural characteristics. The extracted features are normalized in order to make the word representations become insensitive to variations in font, style, size and various image degradations. Then, a novel DTW based matching scheme is employed to take care of morphologically variant words. The proposed system supports cross-lingual search.

Furthermore, some query words are synthetically created before being processed in the word retrieval process. In (K. Zagoris, N. Papamarkos, C. Chamzas 2006), a query word is generated using Arial font. A fixed length feature vector is defined for each segmented word using six scalar characteristics and three initial coefficients of the discrete cosine transform of the profile features. The matching step uses the Euclidean distance on the two feature vectors. Finally, the authors have created a web interface as an experimental platform that can be found at the website[1].

(A. Bhardwaj, D. Jose and V. Govindaraju 2008) proposed a method for script independent word spotting in multi-lingual handwritten and machine printed documents. In this work, a template is created and is used to generate a query word image corresponding to the query text. In fact, this template stores the mapping between a word image and its equivalent text. The system returns a ranked list of word images based on a cosine similarity metric with the query word. A feature vector consisting of 30 geometrical moment values, obtained by moment extraction up to the $7^{th}$ order from the normalized word images, is stored in the main index. Indeed, Moment based features are image scale and translation invariant, which makes them suitable for font independent feature analysis. The authors evaluated their method on a set of documents in three scripts, English, Hindi and Sanskrit. Experiments through these different sets show that the proposed method works well for machine printed text as compared to handwritten and that using Moment based features allows generating better results than using the Gradient, Structural and Concavity (GSC) or Gabor based features. Finally, the authors showed the effectiveness of using statistical Moment based features as opposed to some of the structural and profile based features which may constrain the approach to few scripts.

A contemporary font to generate synthetic character images using a specific font and size is used in (N. Doulgeri,E. Kavallieratou 2009). The query word given by the user in ASCII is transformed in an image word by combining character images with a space between two character images. These character images are obtained by manually writing the different alphabet character by a user and transforming them into a bitmap representation. Global shape features are used to describe the words.

[1] http://orpheus.ee.duth.gr/irs2.

The different extracted features proved to capture the best description of the shape of the word. A smoothing stage follows the feature extraction step in order to deal with small variations due to different fonts and styles of printing as well as possible noise remaining in the word image. Then, the generated feature vector is normalized. In the matching step, a Manhattan Distance is used as a distance criterion. A set of 10 pages containing around 2013 words is used in the evaluation process. The results are given for the 5 fonts (Arial, Courier, Helvetica, Palatino linotype, and Times New Roman) used in the query word synthesis. Indeed, the system performs well when using Times New Roman fonts and a resolution of 300 dpi.

Recently, in a QBS word spotting method (D. Aldavert, M. Rusiñol, R. Toledo and J. Lladós. 2013), the word images are represented by textual and visual representations. In the textual representation, the authors used n-grams of characters bloc to represent word transcriptions. In fact, these transcriptions are divided into a set of bi-grams and tri-grams blocs which are overlapping blocks of individual characters. By accumulating the occurrences of each n-gram into a histogram and normalizing this histogram by its L2-norm, the textual descriptors are generated. Besides, the visual representation is based on a bag-of-visual-words scheme powered by gradient features. The author's contribution is to add spatial information by means of the spatial pyramid proposed by (S. Lazebnik, C. Schmid, and J. Ponce 2006). After that, they used a Latent Semantic Analysis (LSA) algorithm to bring together both textual and visual representation into a common representation space. This is done in order to retrieve the segmented word images which are represented only by visual descriptors. In the spotting process, a cosine distance between the projected query and the projected visual descriptors is used. Experiments are done on the George Washington database where around 4864 segmented words were used. Qualitative and quantitative results are conducted and show this method outperforms some word spotting algorithms such as (Y. Liang, M. C. Fairhurst, R. M. Guest. 2012) (A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012) (V. Frinken, A. Fischer, R. Manmatha, and H. Bunke 2012).

#### 3.1.2.2.2. **QBS guided approaches**

A keyword-guided word spotting in historical printed documents using synthetic data and user feedback is described in (T. Konidaris, B. Gatos, K. Ntzios, I. Pratikakis, S. Theodoridis, S.J. Perantonis. 2007). This work is basically based on the work proposed in (B. Gatos, T. Konidaris, K. Ntzios, I. Pratikakis, S. J. Perantonis 2005) where an approach for keyword search in historical typewritten documents combining image preprocessing, synthetic data creation, word spotting and user's feedback technologies is described. The synthetic query word is generated from the manually selected prototype characters in Greek historical documents. The spacing between characters has been set to 10% of the average character height in the images (see figure 6). Then, the query words are normalized to fit in a pre-defined bounding box. The words in the document collection are segmented using dynamic parameters. For each segmented word, two types of features are defined. In the first type, the areas formed by the upper and lower profiles of the word are calculated in 30 small zones each. In the second type, the image is divided into a set of 90 zones where density of character pixels is calculated. In the matching step, a Manhattan Distance is used between the features of the two words to be matched. An initial list of ranked results is generated improved with a user feedback. The evaluation is done on a sample of 100 document images. 25 randomly selected keywords are used. Experiments show that the combination of synthetic data and user feedback in a hybrid fashion leads to an improved performance for a keyword-guided word spotting system. Here, a word image is built in a non-automatic way; the user is involved to extract an occurrence of query word characters.


**Figure 6: Synthetically generated query word (T. Konidaris, B. Gatos, K. Ntzios, I. Pratikakis, S. Theodoridis, S.J. Perantonis. 2007).**

One year later, in a similar fashion (T. Konidaris, B. Gatos, S. Perantonis,A. Kesidis 2008) proposed a keyword-guided image matching in digitized machine printed historical documents. The word matching process uses four different sets of features, based on (T. Rath, R. Manmatha 2003). These extracted features are then matched using DTW. Before the extraction process, a pruning step is done. It consists in decreasing the number of candidate words by using word portions located at the beginning and end of each word. Indeed, word length is also used. This step plays an important role in improving the obtained results in term of time and efficiency.

Similarly, a word spotting-guided framework for accessing the content of Greek historical machine-printed documents is described in (A. L. Kesidis, E. Galiotou, B. Gatos and I. Pratikakis. 2011). This approach is based on the QBS formalism to generate the query image. Word segmentation is accomplished with the use of two complementary segmentation methodologies, RLSA algorithm and a projection profiles based algorithm, which enhances the performance of the proposed approach. The feature extraction step consists of two distinct phases. First, a normalization process allows preserving scale invariance. Second, two different types of features are extracted, the density of the character pixels and the upper/lower profile projections. Word matching step is based on a Euclidean distance in order to rank the comparison results. Thus, an initial list of results is produced that sorts the segmented word images of the document according to their similarity to the template. In the user's feedback phase, the user selects one or more words in the initial list; they are used in a new matching process. The segmented words are ranked according to their similarity to the selected words. The user intervention increases the accuracy of the retrieval results in terms of recall and precision.

Recently, (M. Keyvanpour, R. Tavoli, S. Mozaffari 2013) introduced a keyword spotting system based on relevance Feedback to improve the accuracy of Document Image Retrieval System. Some of strategies for positive and negative feedbacks such as "Only Positive Feedback", "Only Negative Feedback", and "Positive and Negative Feedback" are compared. It has been shown that the use of a positive Feedback strategy like "Only Positive Feedback" outperforms the Document Image Retrieval Systems.

Besides, we can find in the literature some holistic word spotting techniques that do not query by example or even by string. As it is described in (J.A. Rodríguez-Serrano, F. Perronnin 2009) the authors query by first selecting one or multiple examples from a word and then train a probabilistic model for it. That is what they called "a word class" based query. This approach detects only a set of known query words existing in scanned letters. The authors proposed to use two types of HMMs (Hidden Markov Model), the Continuous HMM (C-HMM) and the Semi Continuous HMM (SC-HMM) for the word model and the Gaussian Mixture Models (GMMs) for the score normalization. The word candidates are detected by evaluating the posterior probability of the candidate given the

model. For the evaluation step, the authors performed experiments on a data set of 630 scanned French handwritten letters where the writing is unconstrained and the letters contain a lot of artifacts. After segmenting the document image into words, the 10 most frequent word classes are used. Experiments show that the SC-HMM gives better results when labeled data is insufficient thanks to the prior information which can be incorporated in the shared set of Gaussians. Furthermore, by using the posterior probability in the matching step, this method extended the query-by-example methods and expects superior performance. Besides, as it trains the model only by using keyword samples instead of training sub-word models, this method can be easily extended to other language. This is considered as an advantage with respect of the existing query-by-string methods.

Finally, we should mention that there are big efforts and dedication in the holistic word spotting community to introduce approaches that can accomplish both QBE and QBS spotting. For this, (J. Almazán, A. Gordo, A. Fornés, E. Valveny 2013) (J. Almazán, A. Gordo, A. Fornés, E. Valveny 2014b) have proposed, lately, a multi-writer word spotting and recognition framework that is based on a Pyramidal Histogram of Characters (PHOC) for representing both word images and strings. This PHOC is learnt by using the powerful Fisher Vector representation of the images. The matching process is reduced in this work to a Nearest Neighbor problem. The experiments for word spotting task are done on George Washington[2], IAM[3], IIIT 5K-word datasets (A. Mishra, K. Alahari, C. V. Jawahar 2012), and The Street View Text (SVT) dataset (K. Wang, B. Babenko, S. Belongie 2011). Another contribution given by this framework is the ability to embed the processed word images and their textual transcriptions into a discriminative space.

# 4. ANALYTICAL ANALYSIS TECHNIQUES

Analytical analysis allows segmenting the word images or even a whole document image into smaller units that will be recognized either isolated or when grouped. Indeed, three analytical techniques categories can be found out in the literature. Some segmentation based approaches require that each word has to be segmented into characters. The crucial step is to split a document image into individual characters (B. Gatos., N. Papamarkos, C, Chamzas 1997). However, the segmentation process cannot be achieved accurately. To overcome this, a lot of works consider multiple segmentation hypotheses by over segmenting images into small units such as connected components, strokes, etc. Other approaches use explicit word segmentation to break segmented words into smaller units that are supposed to be characters that will be recognized after (Y. Lu, M. Shridhar 1996). These approaches can also be divided into two sub-classes depending on the query formulation as the holistic word spotting approaches.

Next, we will give an overview of some analytical based word spotting approaches based either on QBE or QBS formalism. Figure 7 shows the flowchart of the analytical analysis techniques.

[2] http://www.fki.inf.unibe.ch/databases/iam-historical-document-database/washington-database
[3] http://www.fki.inf.unibe.ch/databases/iam-handwriting-database

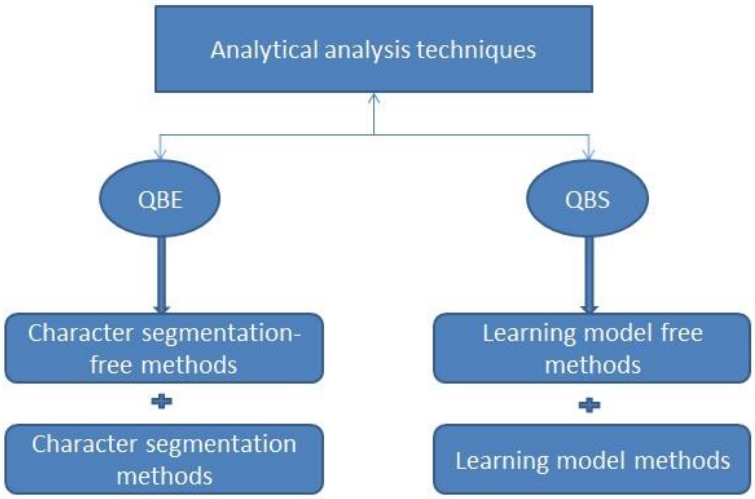


**Figure 7: Flowchart of the analytical techniques.**

### 4.1.1. QBE approaches

We can classify the Analytical QBE based approaches into two main categories: Character segmentation-free based techniques and Character segmentation based techniques. In each category, we will give a brief description of some techniques belonging to it.

#### 4.1.1.1. Character based segmentation free methods

Most techniques of this category use sliding windows in order to extract different features. Figure 8 shows sliding window for extracting small portions for which the different features are computed.

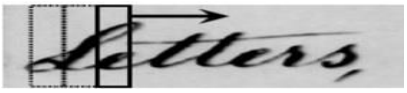

**Figure 8: The process of using a sliding window for extracting the different features from different fragments.**

In (P. Bilane , S. Bres , K. Challita , H. Emptoz 2009), the authors described a word spotting approach based on selective sliding window technique. This method is fully independent of character segmentation. In fact, the size of the sliding window is chosen according to the thickness of the handwritten and to the average height of a text line. Unlike the work of (K. Terasawa, and Y. Tanaka 2007) where all the sliding windows are used, the authors here proceed to the elimination of some

sliding windows that do not respect some criteria. First, regions of interest where possible occurrences of the query word might be located are pre-computed. Then, within those regions, the saliency coefficients of directional roses are extracted. Each retained window is divided into 12 sub-windows and the autocorrelation function is computed in each of them. So, each sub-window is then represented by a signature of 8 values, which result in a feature vector of 96 values for a given sliding window. Finally, Matching between features is done using Euclidean distance correspondence. For experiments, two sets of Syriac manuscripts are used. The first one contains 1813 words taken from a 10 page booklet while the second one contains 1524 words taken from a 4 booklet. The generated results are considered good for indexing purposes even if they include some false occurrences.

Another word spotting method based on line-segmentation, sliding window, continuous dynamic programming and Slit Style Histogram of Oriented Gradient (SSHOG) feature is given in (K. Terasawa and Y. Tanaka 2009). This approach does not require any language-dependent process and it is applied on manuscript documents. A narrow rectangular sliding window is applied on each text line which slides along the writing direction. For each sub-image clipped by the window, a Histogram of Oriented Gradient (HOG) feature vector is calculated. HOG computes a histogram of gradient orientations in a certain local region with the orientation bins evenly spaced over 0-360 degrees. HOG features have the advantage that they deal with many kinds of deformations, variability in the styles, etc. Then, each slit window is divided by using a block optimization into small blocks which overlap in vertical direction and Gradient features are calculated at block level. In the matching step, query word is searched by using the look up table technique and matching the feature vectors of the windows using DTW. For experimental test, the authors used Japanese manuscript images from the scanned diary of "Akoku Raishiki" and an English manuscript from the George Washington Collection. For evaluating, the authors used the Recpret tool cited in (K. Terasawa,H. Imura, Y. Tanaka 2009). Finally, even if there exists a large amount of redundancy in using block features, these redundancies are able to improve the word recognition rates generated by this approach.

Similar work, based on HOG descriptors, is presented recently in (J. Almazán, A. Gordo, A. Fornés, E. Valveny. 2012) and (J. Almazán, A. Gordo, A. Fornés, E. Valveny 2014a). In this framework, the document images, both handwritten and machine printed documents, are divided in equal size cells. Each cell is presented with HOG descriptors. In other terms, the different documents are represented with a grid of HOG descriptors, and a sliding window approach is used to locate the document regions that are most similar to the query. The window size is adjusted to the query size. The authors used a similar approach to the Support Vector Machine (SVM) framework of (T. Malisiewicz, A. Gupta, and A. Efros. 2011) (A. Shrivastava, T. Malisiewicz, A. Gupta, A.A. Efros. 2011) in order to produce a better representation of the query in an unsupervised way. Then, the document descriptors are pre-computed and compressed with Product Quantization (PQ). This approach offers two main advantages, a large number of documents can be kept in RAM memory at the same time and the sliding window becomes significantly faster since distances between quantized HOG descriptors can be pre-computed. The authors evaluated the proposed technique on about 40 pages, handwritten pages from the George Washington data set and typewritten pages from the Lord Byron data set. From their experiments, the authors observed no huge difference in running time on the two data sets. Besides, the accuracy of the method is significantly improved when the cell size is decreased. This is due to the fact that more information can be used when we have more cells in a query. But, increasing the number of cells requires more memory. The highest accuracy is achieved by using small cell size and not the PQ step. However, to take into consideration memory and time constraints, larger cell size and PQ have to be used. The method has some failure situations as in the case of retrieving substrings from long words or giving too much weight to artifacts leading the approach to have confusions with similar shaped words.

An unsupervised analytical using sliding windows for HOG features extraction is presented in (G. Khaissidi, Y. Elfakir, M. Mrabti, M. El Yacoubi, D. Chenouni, Z. Lakhliai 2016). This approach is based on a PQ technique for encoding the extracted HOG descriptors and an SVM classifier with linear function for the recognition process. The approach is evaluated on Ibn Sina handwritten manuscripts dataset which consists of Arabic handwritten documents. The obtained mean average precision rate is of 68.4%.

Moreover, a scalable approach for graphical pattern spotting for historical documents is presented in (S. En, C. Petitjean, S. Nicolas, L.Heutte 2016). The originality of this work is the use of the Vectors of locally aggregated descriptors (VLAD) and Fisher Vectors for feature extraction process. The PQ and asymmetric distance computation (ADC) techniques are used to make the approach scalable. The evaluation of the approach is done on an historical document images[4] consists of 1597 medieval manuscripts documents. These documents represent some constraints such as the degradation, various artifacts, pepper noise, color bleeding. This approach is adaptable to perform the word spotting problem as the graphical spotting problem. The authors have evaluated the word spotting process on the ICDAR 2015 keyword spotting challenge dataset and the overall evaluation metrics are mentioned in[5].

Instead of using a sliding window technique to extract the features, some researchers use block-based feature extraction techniques as in (B. Gatos and I. Pratikakis. 2009). This method is applied to historical printed document. Applying different rotation and scaling variations, 15 query instances for each word query are obtained. Indeed, for each obtained query word, five different sets of feature vectors are generated. Regions of interest in test image which refer to text lines in the image are found by using RLSA in horizontal direction. The matching process is constrained only on certain regions of interest. The query word is compared with rectangular text areas by moving a rectangular window over the region of interest in the test image. A distance measure is used to compare the feature vectors of the query word and tested rectangular area. If several intersecting rectangular areas correspond to successful matching results then authors select the rectangular area with the lowest distance value. This method is based on block-based document image descriptors which satisfies invariance towards translation, rotation and scaling. The evaluation is carried out on a 18$^{th}$ century historical book, from Eckartshausen at the Bavarian state library, where five query words were manually selected and searched for in 100 document images. The recall rate is of 93.2% while the precision rate is of 75.1%.

Furthermore, some techniques perform spotting process by first extracting informative parts in the document images (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007) and (Y. Leydier, F. LeBourgeois, H. Emptoz 2005). In (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007), the authors process medieval manuscripts of Latin and Semitic alphabets. The main idea is to compare only informative parts of the template keyword with only parts of the document that may contain information. The approach is based on differential features that are compared using a cohesive elastic matching method, based on zones of interest (ZOI). In fact, in their context, the best feature to describe the medieval manuscripts is the gradient orientation around high-magnitude zones. Indeed, the proposed matching algorithm is found out to be robust to the horizontal and vertical irregularity of the text. The processed document images are noisy and contain stains, ink dimming, blur, etc. This does not influence the performance of the method which looks to be very robust to all the present degradations. Indeed, this method is also robust to the geometrical distortion occurred during the digitization process. Finally, we conclude that using gradient features with cohesive elastic matching plays an important role to improve the word spotting process.

[4] http://www.docexplore.eu/
[5] http://transcriptorium.eu/~icdar15kws/index.html

In addition to that, in (M. Rusiñol, D. Aldavert, R. Toledo and J. Lladós. 2015) and (M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós 2011) document images are split into a set of overlapping local patches to extract feature vector descriptors. The method here combines the use of a bag-of-visual-words (BoVW) model based on SIFT descriptors and the refinement of the descriptors by using the latent semantic indexing technique. A final voting scheme aims to locate the zones within document images where the queried word is likely to appear. Indeed, this work addresses the segmentation problem by representing regions with a fixed-length descriptor based on the bag-of-visual-words framework. The comparison of regions is much faster since a dot-product or Euclidean distance can be used which makes a sliding window over the whole image feasible. In order to improve the system, the authors used an unlabeled training data to learn a Latent Semantic Indexing space (LSI), where the distance between words is more meaningful than in the original space. So, the LSI is used to learn a latent space where words and documents are more related. The problem is that learning a semantic space with LSI may be too conditioned to the words used in the unsupervised training stage, and adapting to new, unseen words may be complicated. The main important drawback of this sliding-based method is the cost of re-computing the descriptors of every image for every new query. In the evaluation stage, the authors consider different data sets such as the George Washington data set, the Lord Byron data set and a Persian data set. Quantitative evaluation of the method is computed by the mean average precision in the George Washington database and which is of 30.42% where the mean recall was 71.1%. For the Lord Byron data set, the mean average precision was 42.83% and the mean recall was 85.86%. This study is made only for these two data sets because each of them has a ground truth containing the word transcriptions and their bounding-boxes. Indeed, the mean average precision and the mean recall increase significantly depending on the length of the queried word. The larger it is, the higher the rates are. The main advantage of this method is that it can be used for heterogeneous document collections and does not require any preprocessing step as noise removal or normalization.

Recently, (M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós 2011) (L. Rothacker, M. Rusinol, G. A. Fink 2013) (L. Rothacker, L. Vajda, G. A. Fink 2012) have extended the previous work presented in (M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós 2011) by feeding an HMM proposed in (L. Rothacker, L. Vajda, G. A. Fink 2012) with a Bag-of-Features representation powered with SIFT descriptors. This is the first time a Bag-of-features and HMMs are used together in a segmentation-free word spotting application. Mean precision and mean recall score in the GW dataset are improved for short words. Only a single example of the query provided by the user is used to estimate a query model. Thus, it is considered as a soft approach when dealing with a lot of training data.

A similar work to (L. Rothacker, M. Rusinol, G. A. Fink 2013) is described in (L. Rothacker, G. A. Fink, P. Banerjee, U. Bhattacharaya, B. B. Chaudhuri 2013). This work has been evaluated on a printed Bangla dataset. The authors compared their approach to the state-of-the-art results on handwritten Roman dataset as GW dataset with using the same parametric configuration and achieved better results than (J. Almazán, A. Gordo, A. Fornés, E. Valveny. 2012).

Another segmentation-free QBE word spotting approach using Bag-of-features HMMs to model query words is presented in (A. G. Fink, L. Rothacker, R. Grzeszick 2014). The authors extracted the features by SIFT descriptors and the matching process is done by a patch-based approach. This work has been applied on German feldpost postcards dataset considered as a part of a private collection of postcards from World War I (B. Bley, 1917).

Furthermore, two segmentation free keyword spotting methods for Bangla and Roman handwritten documents based on Heat Kernel Signature (HKS) are introduced by (X. Zhang, C. L. Tan 2013) and (X. Zhang, U. Pal, C. L. Tan 2014). The authors extract HKS features from local patches centered at generated Key-points. They used the HKS features instead of SIFT descriptors because it has been proved in (F. Moreno-Noguer 2011) that HKS descriptors perform better than the SIFT ones in

handwriting schemes. Indeed, HKS descriptors tolerate variation better than SIFT ones even in the case where the locations of the similar key points are slightly different. Experiments on the GW dataset were compared with (M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós 2011), the mean precision and recall rates are respectively 62.47 % and 92.38%. Besides, for the Bangla dataset, they are 77.2% and 94.8%.

Moreover, an analytical keyword searching approach based on geometric features is described in (R. Saabni and J. El-Sana 2008). The approach is applied on Arabic handwritten historical documents. These writings are characterized by cursiveness and similarity among letters. First, the authors extracted connected components from the binarized document images; these are classified into main and secondary components based on their size and location with respect to the base line. The main component is the word-part and the secondary components refer to additional strokes which can be associated with a main component. Then, several features capturing the structures of the Arabic script are extracted, with local, semi-global and global behaviors. They refer to segment's angles and length of the word-part's contour. In the matching phase, the authors applied a HMM and a DTW technique. They have slightly modified the DTW technique to include different insertion, deletion and subtraction of segments costs. The evaluation stage is done on three types of Arabic documents, printed documents with different fonts, handwritten documents written by different writers and historical documents. The authors have used 40 pages containing about 8000 words and more than 15000 word-parts. Experiments show that using DTW is better than using HMM in Arabic historical documents. This is due to the fact that it is not always possible to provide enough samples to train a probabilistic classifier in such types of documents.

Lately, an application-independent for spotting queries in document images is described in (H. Chatbri, P. Kwan, K. Kameyama 2014). The contribution is its ability to spot any type of query (i.e. word query or equation query). The authors used the Contour Points Distribution Histogram (CPDH) shape descriptor (X. Shu, X. J. Wu 2011) for feature extraction. They evaluated their work on a dataset of the journal "Annales de l'insée" ("Annales de L'insée" 1980) containing 104 document images including text and mathematical expressions. Three queries were used in the experiments. The average recall value that is more than 70% shows the efficiency of the approach in the spotting task.

#### 4.1.1.2. Character segmentation based techniques

Some researchers have proposed analytical QBE based approaches for Chinese and Japanese document images as in (Y. Lu, C.L. Tan 2004) (K. Terasawa,T.Nagasaki,T.Kawashima. 2005) (Y. Xia, Z.B. Yang, K.Q. Wang 2014).

In (Y. Lu, C.L. Tan 2004), the template image of each character is generated from its standard bitmap image with size 16*16. The authors extract the Chinese characters. All connected components are generated in the document image and then a pruning step eliminates connected components with small areas. Connected components are then merged to generate bounding boxes of the different characters. A black-list is constructed to recode characters that are unsuccessfully bounded. The candidate character matching process contains two stages. A Coarse matching algorithm is carried on the stroke density features to reduce the matched characters set. A weighted Hausdorff distance is aimed to choose the best matching characters. Both algorithms are implemented at character level. The evaluation is done on 132 scanned images taken from recent and traditional Chinese documents digitized at 300 dpi. A set of 128 words of 2-6 Chinese characters is chosen as queries. Obtained results show that this method performs well with an average precision rate of 84.38% and an average recall

rate of 87.74%. Furthermore, the precision rate increases and the recall rate decreases when the word length increases. For instance, for a word with 2 characters, the recall rate is 97.62% and the precision rate is 66.1%. However, when the word consists of 6 characters, the recall rate decreases to 76.9% and the precision rate increases to 100%. The main advantage of this approach is its capability to accurately search query words in Chinese documents in either horizontal or vertical text line, or even both.

In (K. Terasawa,T.Nagasaki,T.Kawashima. 2005), the approach is applied on handwritten manuscripts of Japanese and some other eastern languages. The document image is segmented into small slits, narrow rectangular windows that scan image along the line axis. For each slit, a low dimensional descriptor is generated by Eigen space method. The width of the slits is set to 9 pixels since it should be sufficiently narrow relative to the size of single character the average size of which is observed to be about 60 pixels in their study. The matching process between the low dimensional representation of the query word and the ones of the document images is done using DTW. The advantage is the generation of low dimensional descriptors for each processed image which allows improving and solving the matching problems easily.

In (Y. Xia, Z.B. Yang, K.Q. Wang 2014), the authors used a novel Elastic Histogram of Oriented Gradient (EHOG) feature for calligraphy word spotting. The EHOG is seen as a modification of the known HOG as it takes into account the different characteristics of the Chinese calligraphy character. The experiments are done on 14302 characters within a calligraphy documents dataset. The results show that the EHOG is very effective in the description phase. In the matching phase, the authors have extended the Dynamic Time Warping (DTW) to a Derivative Dynamic Time Warping (DDTW). The DDTW takes into account the feature shape of Chinese characteristic in the matching process.

An analytical QBE based approach is proposed in (H. Cao and V. Govindaraju. 2007). The document images processed here are very noisy. The authors extracted Gabor Features (i.e. Gabor filter and Gabor wavelet) from grayscale character images. Then, the matching process is performed at character level. Similarities based on Euclidean distance and Posterior probabilities are used for matching query word and document words. In a set of 12 medical form images, 5295 character images were taken from the first ten document images for the training test while the last two images are used in the spotting test step. The obtained results show that this approach is time consuming and cannot be extended to larger data sets. Besides, the Posterior probability outperforms the Euclidean distance in this type of documents.

Other researchers have proposed analytical QBE based approaches for Arabic document images as in (M. Kassis, J. El-Sana 2014). In this work, the authors use a radial descriptor to extract features from word-parts images. A radial descriptor defines the intensity variance of the neighborhood of any point at multiple levels. The experiment is done on a set of Arabic historical word-parts images including multiple instances of different word-parts.

### 4.1.2. QBS based approaches

Most of the approaches in this category use DTW technique, HMMs and NNs models to solve the word spotting process. Thus, we can identify two sub-classes: those based on a learning model and those that are learning-free.

#### 4.1.2.1. Learning model-free based techniques

In this type of approaches, we describe the work presented by (R. F. Moghaddam and M. Cheriet 2009), where Cursive Arabic scripts are processed with no need of any word or line segmentation step, it is language independent. Furthermore, this technique can be applied on very old historical document images and it allows extracting the salient information. Among all the connected components in the document image, a Basic connected component (BCC) library is generated which contains the connected components found within the text. Indeed, this multi-class library is created based on six features as aspect ratio, horizontal frequency, scaled vertical center of mass, number of branch points, height ratio to line height and presence of holes. The clustering process is the SOM. In fact, each connected component is compared to all clusters and the nearest BCCs are selected as matching BCCs of that connected component. The multi-class library is extended by matching the normalized horizontal and vertical histograms of a new CC to the existing ones using DTW. The spotting step is done by using Euclidean distance measure computed by the dynamic time warping technique. Indeed, the authors used a SOM-based technique in order to reduce the computational complexity. The experiments are done on a set of 20 document pages of a very old and degraded text taken from Juma Al Majid Center. The resolution is 150 dpi. The approximate number of CCs is around 11000. For the querying phase, two users were asked to write queries as similar as they can to the original script. Experiments show that this method cannot perfectly separate between dots and CCs, explaining the very low precision rates.

Furthermore, (K. Khurshid, C. Faure, N. Vincent 2009) describes a word spotting system for historical documents based on the extraction of segmented words as well as characters in the text. In the query formulation step, the user can either click on a word in the document image or he can type it as an ASCII word. In this latter case, the sequence of characteristics associated with each character is associated with the word. So, the method is independent of the size of the characters. The matching is performed at two levels, character and word. After the character extraction step, each character is labeled with a feature set consisting of six features sequences and five scalar characteristics. Four of these features are defined in (T. Rath, R. Manmatha 2007). Besides, the scalar features are used for coarse matching and decision making while the vector features are used in a dynamic matching process. Characters are matched by comparing their features sets using DTW while words are matched by comparing the strings of characters using a Merge-Split Edit distance algorithm. The obtained results show that recognition results are very promising. This system does not depend on a perfect character segmentation process which is hard to obtain especially when dealing with poor quality historical document images. The use of the proposed Merge-Split Edit distance makes the system very robust because this technique is able to cater for the segmentation inconsistencies. Besides, defining features at the segmented characters (S-character) level allows giving better representation of words as compared to word level features. This presents another advantage of this system. Moreover, the system is scale and translation invariant thanks to the use of a DTW algorithm to match the S-character features.

In a similar fashion with the previous work (K. Khurshid, C. Faure, N. Vincent 2009), the authors in (Y. Liang, M. C. Fairhurst, R. M. Guest. 2012) construct a feature based representation known as the synthesized word from a query word. Each word image in the documents is presented by graphemes obtained by segmenting the word images into small units. In fact, the synthesized word is a vector of the appropriate character models obtained with a node number associated with the graphemes labeled according to their similarities in an unsupervised learning SOM method. The retrieval

module consists in finding the probability that the K-th character in the test word is an instance of the K-th character in the query word. The mapping of the graphemes maximizes the likelihood between the characters. Then, the retrieved words are ranked according to their associated probabilities with the synthesized word. In the experiment, the handwritten manuscript from "a travel diary " written in 1645 available at the Canterbury Cathedral Archive is a data set which is divided into training and testing sets. The training set contains at least one sample for each of the characters modeling the query word. Results are obtained by evaluating the proposed method on historic and modern manuscripts. The historical documents are from the George Washington data set and the "a travel diary". Besides, the modern documents are collected from a local writer. Finally, the proposed character-based modeling or grapheme spectrum technique and the word modeling or synthesized word technique make this framework able to retrieve out-of-vocabulary words from the document collection, using a small amount of training data contrary to some approaches that retrieve only lexicon words that are encountered in the training set.

#### 4.1.2.2. Learning model based techniques

Some analytical QBS based systems have focused on several variants of Hidden Markov Models (HMMs) or even on Neural Network (NN) to address the word spotting process (A. Fischer, A. Keller, V. Frinken, and H. Bunke. 2010) (V. Frinken, A. Fischer, H. Bunke and R. Manmatha 2010) (F. R. Chen, D.S Bloomberg, L.D Wilcox., 1995) (A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012) (V. Frinken, A. Fischer, R. Manmatha, and H. Bunke 2012) (J.A. Rodríguez-Serrano, F. Perronnin 2009) (S. Wshah, G. Kumar, V. Govindaraju 2012) (S. Thomas, C, Chatelain; L. Heutte; T, Paquet. 2010) (A. H. Toselli, E. Vidal 2014) (J. Puigcerver, A. Toselli, E. Vidal 2016) (J. Puigcerver, A. h Toselli, E. Vidal 2014).

A keyword spotting derived from a neural network based system for unconstrained handwritten documents is described in (A. H. Toselli, E. Vidal, V. Romero, V. Frinken 2013b). The method performs at document level. Splitting text into words is not required. Besides, the authors used an ASCII transcription of the text lines to train the neural network. Recovering word images from the IAM data set (V. Frinken, A. Fischer, and H. Bunke 2010), the authors used a horizontally sliding window with a width of one pixel to extract nine geometric features at each position from left to right, three global and six local ones. The global ones are the three first order moments ($0^{th}$, $1^{st}$ and $2^{nd}$ moments of the black pixels within the window). The local features are the position of the top/bottom most black pixel, the inclination of the top/bottom contour of the word at that position, the number of vertical black/white transitions, and the average gray value between the top/bottom most black pixel. The word spotting process is done by using a Token passing Algorithm in conjunction with BLSTM (Bi-directional long-short-term memory) neural networks. In the experiments, 7 neural networks are trained using a set of 6.161 lines as training set, a set of 920 lines as a writer independent validation set and a set of 920 lines a test set. These sets are taken from the IAM data set that is known for its multi-writers handwritten English document images. The 4000 most frequent words from these sets are used in the keyword spotting process. According to the precision-recall rates, the performances of the networks vary considerably and the best performance reaches an average precision rate of 82.89%. The main advantage of this method is that it enables to search for queries that do not appear in the training set.

An extended work similar to (A. H. Toselli, E. Vidal 2014) and (A. H. Toselli, E. Vidal, V. Romero, V. Frinken 2013a) is presented in (J. Puigcerver, A. Toselli, E. Vidal 2016). This approach

overcomes a major problem of lexicon-based methods that is the out-of-vocabulary queries (OOV). Besides it outperforms the work proposed in (J. Puigcerver, A. h Toselli, E. Vidal 2014).

Some works proposed to use HMMs in line based spotting approaches such as (A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012) (F. R. Chen, D.S Bloomberg, L.D Wilcox., 1995) (S. Wshah, G. Kumar, V. Govindaraju 2012).

(A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012) described an extended version of (A. Fischer, A. Keller, V. Frinken, and H. Bunke. 2010), a learning based word spotting system for unconstrained handwritten text based on character HMMs. The input is an arbitrary keyword string and a text line image. Every input text line image is normalized to cope with different writing styles. Then, the text line images are represented by 9 local features, as in (U.V. Marti,H. Bunke 2001), generated by a sliding window. During the training phase and based on the transcribed text line image, character HMMs are trained for each character of the alphabet. Then, at the matching step, a likelihood score of the input text line, that is going to be normalized, is calculated by connecting the trained character HMMs to a keyword text line model. The experimental evaluation is conducted on three well known data sets. The IAM data set that contains multi-writer modern English texts, the George Washington data set and the Parzival data set that contains medieval manuscripts with a common writing style. Indeed, a set of 3421 non-stop words from 4000 most frequent words are used. All words of the cross-validation training set are used for the George Washington data set and finally all 3220 words from the training set are adopted for the Parzival data set. Experiments show the technique outperforms some standard template matching techniques such as the DTW-based system proposed in (T. Rath, R. Manmatha 2007). The system has some advantages. First, it is a lexicon-free system and such system imposes a high computational complexity. Second, it does not depend on a segmentation process into words at both training and recognition stages. Third, it gives better performance than the learning based systems at word level because it trains only a small number of character classes. Finally, it is able to spot any keywords even if they are not present in the training stage. However, as it is hard to obtain transcribed text line images from historical documents, the system found lot of difficulties in the training stage which would decrease the performance of the system. Also, the drawback of this approach is the large computational cost of the keyword specific HMM Viterbi decoding process that generates the confidence scores of each query.

A similar approach as (A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012) is presented in (S. Thomas, C, Chatelain; L. Heutte; T, Paquet. 2010) where an information extraction system applied on handwritten unconstrained documents is described. At the recognition stage, the acceptation or the rejection is controlled by the variation of a hyper-parameter in the HMM line model.

In addition, (A. H. Toselli, E. Vidal 2013) proposed a HMM-Filler word spotting based approach for handwritten documents. The authors built only a “Filler” model to compute the confidence score of each query. They overcome the computational cost drawback of the work presented in (A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012).

Another work presented in (F. R. Chen, D.S Bloomberg, L.D Wilcox., 1995) has the ability to outperform the line based approach presented previously. The authors proposed a script independent line based word spotting in handwritten documents based on HMMs. The keywords are presented as a sequence of character models in model space and the filler models are used for a better representation of non-keyword text. A reduced lexicon is used to overcome the high computational complexity resulting from using all the non-keywords. The evaluation phase is conducted on English, Arabic, and Devanagari documents. One of the advantages of the system is that it is able to deal with large vocabulary without any word or character segmentation process.

The authors in (J.A. Rodriguez-Serrano, F. Perronnin 2009) proposed a similar gradient-feature based approach as in (K. Terasawa and Y. Tanaka 2009). Modern fonts are used to create synthesized

queries for matching with handwritten text. Local Gradient Histogram known as LGH (J. Rodríguez-Serrano and F. Perronnin 2008) based features are computed to encode word shapes robustly. Besides, a sliding window is used to divide word image into overlapping windows further divided into 4*4 grids. For each of these 16 cells, a gradient histogram is calculated. Thus, 16*8 features are defined for each window (see Figure 9). In the matching step, a semi continuous Hidden Markov Model (SC-HMM) is used in which the feature space is clustered using a Gaussian mixture model (GMM). In fact, the GMM is trained offline with a large set of LGH feature extracted from many windows in the tested images. For the synthetic word queries, an LGH feature sequence is extracted for each word and SC-HMM is trained using these sequences. The method is validated on a data set of 105 scanned letters written in French which are provided by a company. This data set is characterized by the variability of writers, styles, artifacts and spelling mistakes. Besides, they train a GMM with 512 Gaussians by using approximately 1,000,000 feature vectors extracted from set of letters. This GMM is used for training all the 10 states per character SC-HMMs. Obtained results in terms precision/recall plot, show that the proposed method has a competitive performance.

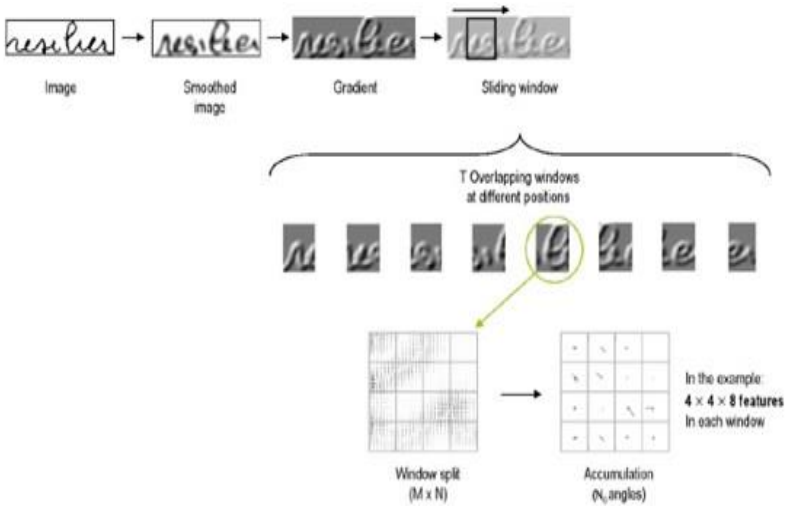


**Figure 9: The process of LGH extraction (J.A. Rodriguez-Serrano, F. Perronnin 2009).**

Recently, based on the previous work of (J.A. Rodriguez-Serrano, F. Perronnin 2009), the same authors (J. Rodríguez-Serrano and F. Perronnin 2012) proposed a novel similarity measurement between vector sequences. Each sequence is first mapped to a HMM and then a measure of similarity is computed. Indeed, the authors proposed to model sequences with semi-continuous HMMs (SC-HMMs). The advantage is that the computation of a similarity between two SC-HMMs can be simplified to a DTW between their mixture weight vectors, which significantly reduces the computational cost. Experiments are carried out on three different data sets which are a set of real handwritten letters, the George Washington data set and the IFN/ENIT data set[6] of Arabic handwritten words. The conclusion made by the authors is that the proposed similarity gives better results than the traditional DTW when it is applied between the model-based approach using ordinary C-HMMs and the original sequences. Indeed, this novel similarity leads to a good accuracy.

[6] http://www.ifnenit.com/

After describing and detailing most of the existing spotting based methods, we are now going to give a brief conclusion about the different scripts, features and also the several matching techniques used in these word spotting approaches.

# 5. CONCLUSION AND REMARKS

Here, we will summarize the important features used in different word spotting approaches and enumerate the different methods that have been used in the spotting process.

First, in order to give the readers a clear overview of the word spotting tasks, we would like to mention in table1 the different categories and sub-categories of main existing word spotting approaches and enumerate their advantages and drawbacks. Then, for completeness' sake, we would like to guide readers within lot of constraints and help them to overcome them.

**Table 1: The different categories and sub-categories of main existing word spotting approaches**

| Class | Sub-class | Publications | pros | cons |
|---|---|---|---|---|
| Holistic | QBE | (T. Rath, et al. 2007) (T.M. Rath, 2003) (M. Rusinõl et al. 2008) (J. Li et al. 2009) (D.A. Lisin et al. 2005) (A. Kołcz et al. 2000) (A. Andreev et al. 2009) (J. Lladós, et al. 2012) (T. Rath et al. 2003) (B. Gatos et al. 2009) (S. Bai et al. 2009) ( P. Wang et al. 2014) (R. Shekhar et al. 2012) (V. Lavrenko et al. 2004) (H. A. Glucksman et al. 1967) (I.Z. Yalniz et al. 2012) (M. Rusinõl et al. 2014) (P. Keaton et al. 1997) (J. Almazãn et al. 2012) (D. G. Long et al. 1981) | • Different information about document images are provided.<br>• Robustness toward the words length.<br>• Robustness towards occlusions and image deformation.<br>• Invariant to illumination changes and image noise.<br>• Different information about convexities and concavities.<br>• Enables to deal with occlusions, rotations, transformation, etc. | • Word segmentation process.<br>• The query must be present in processed documents. |

| | | | | |
|---|---|---|---|---|
| | QBS | (S. Marinai et al. 2006) (S. Marinai,et al. 2005) (S. Marinai et al. 2006) (S. Marinai et al. 2007) (S. Marinai et al. 2004) ( B. Gatos et al. 2005) (A. Balasubramanian et al. 2006) ( K. Zagoris et al. 2006) (T. Konidaris et al. 2007) (A. Bhardwaj et al. 2008) (N. Doulgeri et al. 2009) (J.A. Rodríguez-Serrano et al. 2009) (A. L. Kesidis et al. 2011) ( Y. Liang et al. 2012) (D. Aldavert et al. 2013) (M. Keyvanpour et al. 2013) (J. Almazán et al. 2013) (J. Almazán, et al. 2014) | • The query may not be present in processed documents.<br>• May have the same pros as holistic QBE techniques. | • Word segmentation process.<br>• Not reliable in historical documents.<br>• Generation of synthetic images. |
| Analytical | QBE | (Y. Lu et al. 2004) (K. Terasawa, et al. 2005) (Y. Leydier et al. 2007) (Y. Leydier et al. 2005) (W. Pantke et al. 2013) (K. Terasawa et al. 2007) (R. Manmatha, et al. 1996) (R. Saabni et al. 2008) (P. Bilane et al. 2009) (K. Terasawa et al. 2009) (S. Marinai et al. 2003) (J. Almazán et al. 2012) (L. Rothacker et al. 2013) (L. Rothacker et al. 2013) ( X. Zhang et al. 2013) (X. Zhang et al. 2014) (M. Rusiñol et al. 2015) (M. Rusiñol et al. 2011) (A. G. Fink et al. 2014) (H. Chatbri et al. 2014) (Y. Xia et al. 2014) ( M. Kassis et al. 2014). | • No segmentation process in some cases.<br>• Applied on all types of documents.<br>• Robust to degradations, geometric distortion, etc. | • Character segmentation process.<br>• Time consuming. |
| | QBS | (A. Fischer et al. 2012) (V. Frinken et al. 2012) (F. R. Chen et al. 1995) ( R. F. Moghaddam et al. 2009) (J.A. Rodriguez-Serrano, et al. 2009) (K. Khurshid et al. 2009) (A. Fischer et al. 2010) (V. Frinken et al. 2010) (S. Thomas et al. 2010) (S. Wshah et al. | • May have the same pros as Analytical QBE techniques.<br>• Learning models for better spotting rates. | • Time consuming in the offline learning process. |

| | | | | |
|---|---|---|---|---|
| | | 2012) (A. H. Toselli et al. 2014) (A. H. Toselli et al. 2013) (A. H. Toselli, et al. 2013°) | | |

Second, feature extraction phase depends greatly upon the image nature and image representation. In the literature, most of the researchers based their works on the features extracted from binary images. However, we found out that most of words spotting approaches of historical handwritten documents are based on Gray-Scale Images. This is due to the fact that binarization in this type of document results in the loss of important information that leads to unsatisfactory results. Therefore, we think that it is very interesting for the community of word spotting to base their works on images represented by other color spaces when possible. For the image representation, we can find three major representations: Pixels, Skeleton and Contours. In Pixel representation, the feature extraction step depends upon correct estimation of each word's baseline and the quality of the applied image binarization technique. However, this is not required in Contour representation. Besides, image needs to be skeletonized in order to extract some features as for instance end points or intersections. The different features used for document description are mentioned in the following table. The superscripts H, T, and B mentioned after each publication in table2 indicate that the proposed feature extraction technique is used on Handwritten, Typewritten, or on both Handwritten & Typewritten image documents.

**Table 2: The most used word representation features in word spotting based approaches.**

| Publications | Features used | Image format | Representation |
|---|---|---|---|
| **[(A. Kołcz et al. 2000) (A. Andreev et al. 2009) histo, (D.A. Lisin et al. 2005) histo]$^{H}$** | Projection profiles, upper/lower profiles, background to-ink transitions, gray scale variance, partial projection profile, Gaussian smoothing, and Gaussian derivatives. | | |
| **[(J. Lladós et al. 2012) histo]$^{H}$** | Corner features | | |
| **[(T. Adamek et al. 2007) moder]$^{H}$** | Gabor features | | |
| **(F. R. Chen et al. 1995)** | Column Pixel values | | |
| **[ (K. Terasawa et al. 2005) moder]$^{H}$ [ (S. Marinai et al. 2003) (J. Almazán, et al. 2012)]$^{B}$, [ (Y. Leydier et al. 2007) histor]$^{H}$** | Local gradient histogram, histogram of oriented gradient (HOG), Gradient orientation | | |

| | | | |
|---|---|---|---|
| [ (K. Terasawa, et al. 2009) (B. Gatos et al. 2005) hist]$^{T}$, [ (T. Konidaris et al. 2007) hist]$^{T}$ , [ (A. L. Kesidis et al. 2011) histo]$^{T}$, [ (J. Li et al. 2009)]$^{H}$, [ (N. Doulgeri et al. 2009) hist]$^{T}$, [ (A. Bhardwaj et al. 2008)]$^{B}$, [(A. Balasubramanian et al. 2006)]$^{T}$ | Mesh features, upper/lower profiles, density of character pixels, background to-ink transition, height, width, amount of estimated ascenders/descenders, vertical histogram, middle profile, position of the peak of area of ascenders/descenders, Projection profiles, moments, mean black pixel distribution, strokes curvature | Gray Scale | Pixel |
| [(V. Papavassiliou et al. 2010) moder]$^{H}$ | Gradient based binary features (GSC binary features) | | |
| [(R. Saabni et al. 2008) hist+moder]$^{B}$ | Geometric features | | |
| [(R. Shekhar et al. 2012)]$^{T}$ | Local feature sequence | | |
| [ (Y. Lu et al. 2004)]$^{T}$ | Strokes' density | | |
| [(M. Rusinõl et al. 2008) hist]$^{H}$ | Projection profiles, cavity features | Binary | Skeletons |
| [(V. Lavrenko et al. 2004)]$^{T}$ | Character ascenders/descenders, deep eastward/westward connectivity, holes, i-dot connectors and horizontal line intersection | | |
| [(K. Zagoris et al. 2006)]$^{T}$ | Width to height ratio, end points, cross points, image area, center of gravity, horizontal/vertical projections, top/bottom shape projections, upper/lower grid features | | |
| [(H. A. Glucksman et al. 1967) hist]$^{H}$ | Pseudo-structural model based on Loci-feature representations | | |
| [(B. Gatos et al. 2009) hist]$^{H}$ | Multi-scale convexities/concavities features | Binary | Contours |

After reviewing all the researches in the literature of word spotting or information retrieval in document images, we may conclude that the feature extraction techniques that generate good results in one application may turn out not to be successful in other applications. Thus, we would like to advise the community of word spotting and information retrieval to take into consideration all the constraints imposed by their application and especially the processed document images before doing any choice for an implementation. Here, in addition to the information given in Table 1 and Table 2, we would like to point out some necessary recommendations when dealing with word spotting questions.

Generally, the documents can be single-font or multi-fonts typed or unconstrained handwritten ones. To begin with, when we are dealing with multi-fonts documents, we recommend using moment features extraction techniques that are scale and translation invariant. In addition to that, some researches also use Geometric features for recognizing multiple fonts. Nonetheless, Geometric features are not invariant to deformation and not very recommended when dealing with information that have been corrupted. Therefore, to classify mixed font alphabets, pseudo structural based on Loci features can be used. Furthermore, when the processed documents have contrast variations such as faded ink, differences in pressure or even disconnected letters, we can extract features from a single closed contour of each word or character. In addition, Gabor features can be used for this type of limitations. Moreover, to deal with the unconstrained and cursive handwritten forms, global and local features in the form of profile signatures and morphological concavities can be extracted. They are widely used in the context of documents that are written by various writers. In addition to that, we recommend adopting Gradient features such as Histogram of Oriented Gradient (HOG), Slit Style Histogram of Oriented Gradient (SSHOG), and Local Gradient Histogram (LGH) to handle much kind of deformations and variability in the styles.

Within those documents, characters may have different characteristics. They may be self-touching, fragmented or even merged. They also may have broken loops. When dealing with these constraints, techniques that extract features from character contours or skeleton should be used. These techniques are the most used in the literature to handle these variants. Besides, when considering touching handwritten numeral strings, we recommend using the morphological structural features. Additionally, some applications are designed to segment the uppercase serif printed characters, in that case, the quasi-topological and topological features extraction techniques are suggested. Furthermore, in the processed documents, some transformation may occur to the different characters. In fact, each character may be rotated, skewed, scaled, mirrored and translated. To cope with these variations, we advise to use the invariants. For instance, rotation variant features are used to separate between ‘u’ and ‘n’, or between ‘6’ and ‘9’. However, Size invariance is important to distinguish between ‘o’ and ‘O’, or between ‘,’ and ‘9’. In addition, when the tested documents have some slanted fonts or when the characters are more or less slanted, skew invariance is recommended. But, we should say that not all the variance and invariance features are able to model all the variation among same class characters, so, we can adopt the projections histograms in the separation between characters such as ‘m’ and ‘n’, for instance. These latter techniques are very sensitive to variability in writing style. In addition to that, characters like ‘p’ and ‘q’ can be distinguished by using profile features. Besides, Bag of words representation model can be used to deal with character occlusions or image deformations. It is invariant to changes of illumination or image noise.

Finally, it is also recommended to combine several features from different feature extraction techniques in order to achieve better representations of the different image documents. For the sake of completeness, more information about the different classes of the extracted features can be found above in the previous sections.

Furthermore, spotting process or word matching process is considered as a difficult stage in word spotting systems. We can conclude from the previous study that the matching techniques can be divided into two categories, the training based techniques and the training-free techniques. For the first class, we remark that the Hidden Markov Models are widely used. Besides, the Dynamic Time Warping (DTW) and the Euclidean Distance are widely used for the second class.

A summary of different matching techniques used in word spotting approaches is given in table 3.

Table 3: The matching techniques used in main word spotting approaches.

| Publications | Methods used | Category |
|---|---|---|
| (A. Kołcz et al. 2000) (A. Andreev et al. 2009) (J. Li et al. 2009) (B. Gatos et al. 2009) (A. Balasubramanian et al. 2006) (W. Pantke et al. 2013) (R. Saabni et al. 2008) (K. Khurshid et al. 2009) (T. Konidaris et al. 2008) | DTW | Training-free |
| (J. Lladós et al. 2012) (H. A. Glucksman et al. 1967) (K. Zagoris et al. 2006) (A. L. Kesidis et al. 2011) (R. Manmatha et al. 1996) (P. Bilane et al. 2009) (R. F. Moghaddam et al. 2009) | Euclidean Distance | |
| (K. Zagoris et al. 2014) (Y. Lu et al. 2004) | Hausdorff Distance | |
| (R. Shekhar et al. 2012) | Substring Matching | |
| (H. A. Glucksman et al. 1967) | Cosine Distance | |
| (N. Doulgeri et al. 2009) (T. Konidaris et al. 2007) | Manhattan Distance | |
| [(Y. Leydier et al. 2007) | Cohesive Elastic Matching | |
| (K. Terasawa et al. 2009 | Rectangular Window Matching | |
| (Y. Lu, et al. 2004) | Coarse Matching | |
| (V. Frinken et al. 2010) (V. Frinken et al. 2010) | Neural Networks | Training based |
| (A. Fischer et al. 2010) (A. Fischer et al. 2012) (J.A. Rodriguez-Serrano et al. 2009) (J. Rodríguez-Serrano et al. 2012) | HMMs | |

By now, we have discussed two main categories of the word spotting approaches, holistic or segmentation-free techniques and analytical or segmentation-based techniques. Each category can be divided into two sub-classes depending on the formulation query. These sub-classes are the Query-By-Example (QBE) class and the Query-by-String (QBS) class. In the holistic category, the word spotting approaches compare a sequence of observations derived from a query with similar features for the words in the documents and obtain directly good spotting rates. They are mostly used when dealing with poor quality and printed historical document because they do not require character segmentation that is a difficult process in these types of documents. In fact, these documents are very noisy, have irregularity in printing, have different font variations, etc. At the opposite, the analytical approaches tend to match between consecutive sequences of primitive segments or characters of a word. Generally, they are able to focus on the local intrinsic characteristics of words and the matching process is done at low level as character level. These facts make the analytical approaches very robust and give better spotting results with respect to the holistic ones.

During nearly 30 years, plenty of studies have been done for spotting/retrieval information from modern and historical documents. However, needless to say that there is no work that has been able to achieve a good performance and a precision rate acceptable for commercial applications. They are all dedicated to a specific corpus. Hence, lot of research materials need to be done in the different spotting/retrieval information process in order to obtain and create more robust and more efficient systems with improved performance and reduced computational cost.

# 6. TOWARDS THE PROPOSED APPROACH

First of all, from above, we conclude that analytical approaches are recommended to be used when the writings of processed documents have pretty good quality. As a matter of fact, these approaches permit either segmenting the entire documents or words into smaller units that will be recognized when isolated or grouped or permit manipulating globally the entire document without any segmentation process. Since, the word segmentation process is considered as a very difficult step to be done in historical document with poor quality or in modern documents with various written styles, a global word spotting approach that will be applied on the entire document is suggested.

Moreover, in the context of our project, we have chosen to express the query as a sequence of characters constructed by a keyboard input. This authorizes more freedom to the user as the query may be expressed in a way unrelated to the searched documents. Indeed, this would allow our system to be used in all circumstances even if the query does not exist within the manipulated documents.

Furthermore, as our approach is analytical, then the choice of the adopted features must achieve a tradeoff between efficiency and fast computation time. Indeed, these features have to be built according to the way human perception reacts with respect to the shape of a word, focusing for instance on the beginning or the end of a word or on more salient parts like ascenders and descenders. Otherwise, they must be highly depending on the query or document properties, and essentially take into account the computation time. Taking into consideration all these challenges, we have selected Haar-like features as they are simple and efficient. However, we will not only apply the basic Haar-like features, but we will generalize them according to the characteristics of queries and to the manipulated documents. These features will be adopted in both text localization in comics and word spotting in manuscript documents studies that will be reviewed next in this report.

Finally, we aim to avoid a training stage on a database and design a system capable of adaption and self-learning. Besides, we intent to obtain an approach that has to be able to automatically detect some of these characteristics such as the size of written characters for instance. Thus, we do not choose to base our word spotting proposed approach on any learning model based techniques.

After preparing a solid base for our studies, we are now going to detail and deeply describe the different studies that have been done through this thesis.

*"He who is not courageous enough to take risks will accomplish nothing in life"*_**Muhammad Ali**

# III. Haar functions and Haar-like features

## Contents

# 1. INTRODUCTION

After reviewing all the researches in the literature of word spotting or information retrieval in document images, we concluded that the feature extraction techniques that generate good results in one application may turn out not to be successful in other applications. It seems reasonable not to fix *a priori* the different features but adapt them to the query and the processed documents. Yet, we have chosen a set of features that offer several opportunities. We have opted for Haar-like features extraction technique for the proposed word spotting framework in this thesis due to its interesting advantages. For the sake of completeness, we are going to highlight the definition and clarify the two concepts of Haar functions and Haar-like features.

We are going now to describe and detail the discrete Haar transform and the Discrete Haar wavelet transform in image processing and pattern recognition and review the various applications that are based on Haar-like features.

# 2. HAAR FUNCTIONS AND HAAR LIKE FEATURES

This section outlines the concept of Haar transform and Haar wavelet transform (subsection 2.1) and Haar-like features (subsection 2.2).

## 2.1. Haar transform and Haar wavelet transform

Haar functions are first mentioned in the thesis of the mathematician Dr. Alfréd Haar (A. Haar 1910). In the literature, Haar functions are defined by different ways. regarding the generated values of Haar functions at points of discontinuity (A. Fournier 1996) (T. Blu, M. Unser 2002). Basically, the Haar function which is an odd rectangular pulse pair is considered as the oldest and simplest orthonormal wavelet (A. Fournier 1996). Indeed, some modification of Haar functions are introduced in (R. Claypoole, G. Davis, W. Sweldens, R. Baraniuk 2007) (P. porwik, A. Lisowska 2004b). In image processing and pattern recognition, Haar functions are used in edge extraction, image coding, etc. Moreover, they are applied in different applications such as those proposed by (P. Jorgensen 2003) (L. Daubechies, W. Sweldens 1998) .

Haar functions definitions differ regarding the values of Haar functions at points of discontinuity. For instance, (F. H. Harmuth 1978) defines Haar functions as follows:

$$\begin{cases} haar(0,t)=1, \ for \ t\in[0,1[ \\ haar(1,t)=\begin{cases} 1, & for \ t\in\left[0,\frac{1}{2}\right[ \\ -1, & for \ t\in\left[\frac{1}{2},1\right[ \end{cases} \end{cases} \quad (1)$$

Where

$$\begin{cases} haar(k,0)=\lim_{t\to 0^{+}} haar(k,t) \\ haar(k,1)=\lim_{t\to 1^{-}} haar(k,t) \end{cases} \quad (2)$$

And at the point of discontinuity, the value would be:

$$haar(k,t)=\frac{1}{2}(haar(k,t-0)+haar(k,t+0)) \quad (3)$$

However, some authors admit that values of Haar functions at the points of discontinuity are zeros and they adopt the following formula:

$$haar(k,t)=haar(k,t+0) \quad (4)$$

Discrete Haar functions are obtained by sampling the orthogonal Haar functions at $2^n$ points. n here is an integer. Indeed, these function are defined on [0,1[ interval and they are obtained also by the following formula:

$$\begin{cases} H(n)=\begin{bmatrix} H(n-1) & \otimes & [1 \ \ 1] \\ 2^{\frac{n-1}{2}} I(n-1) & \otimes & [1 \ -1] \end{bmatrix} \\ where \\ H(0)=1 \end{cases} \quad (5)$$

In this formula, H(n) represents the discrete Haar functions. This matrix is a non-symmetric matrix of order $2^n$. Unlike the Hadamard matrix, which is a symmetric block matrix with entries of only +1 and -1 and whose rows are mutually orthogonal, the Haar matrix contains entries of +1, -1, and additional zeros if required, multiplied by powers of $\sqrt{2}$ (L. D. Baumert, M. hall 1965) (S. Georgiou, C. Koukouvinos, J. Seberry 2003) (R. K. Yarlagadda, J. E. Hershey 1997) (J. Seberry, B. Wysocki, T.

Wysocki 2005) (S. D. You, W. H. Chen .2013). I(n) is the identity matrix of degree $2^n$ and $\otimes$ is the Kronecker product.

As mentioned above, Haar functions are considered as Wavelets. Yet, a mathematic definition of Haar wavelet goes as follow: "The Haar wavelet is a sequence of square-shaped or rescaled functions that together form a wavelet family or basis".

Haar wavelets are first mentioned in the thesis of the mathematician Dr. Alfréd Haar (A. Haar 1910). Various definitions of the Haar wavelets and generalizations may be found in (R. S. Stankovic, B. J. Falkowski 1997).

As in wavelet analysis, two principle functions define the Haar Wavelet. :

- The scaling function known as the father wavelet which is defined as:

$$\phi(t) = \begin{Bmatrix} 1 & , \ if \quad (0 \le t < 1) \\ 0 & , \ otherwise \end{Bmatrix}, with \ \phi: \square \to \square \qquad (6)$$

- The wavelet mother function which is defined as:

$$\psi(t) = \begin{Bmatrix} 1 & , & if\,(0 \le t < 1/2) \\ -1 & , & if\,(1/2 \le t < 1) \\ 0 & , & otherwise \end{Bmatrix}, with \ \psi: \square \to \square \qquad (7)$$

Figure 10 is a simple illustration of these two functions.

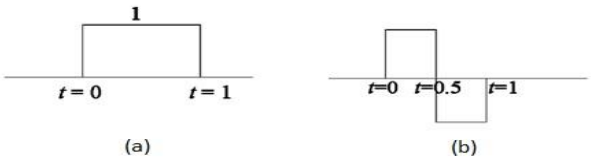


**Figure 10: The two functions that characterize the Haar wavelet. (a) The scaling function. (b) The wavelet mother function**

Furthermore several properties are verified by Discrete Haar wavelet Haar transform (O. S. Jahromi, B. A. Francis, R. H. Kwong 2003). Indeed, taking into consideration the formula (5), any Haar function and scaling function may be computed by the following formulas:

$$\begin{cases} \phi_i^j(t) = \sqrt{2^j}\,\phi(2^j t - i) \\ \psi_i^j(t) = \sqrt{2^j}\,\psi(2^j t - i) \\ \quad where \\ \qquad i = 0,1,2,....,2^j - 1 \\ \qquad j = 0,1,.....,\log_2 N - 1 \end{cases} \tag{8}$$

We mention that generated functions $\phi_i^j(t)$ are orthogonal between them. The same is for $\psi_i^j(t)$ functions. Thus, each generated set of functions is represented in an orthonormal family and it has a compact support. Yet, Discrete Haar wavelet can be defined as a series constituted by different re-scaled square functions that allow representing functions over intervals in terms of decomposition with respect to orthonormal functions. Moreover, Formula (5) and formulas (8) shows that the Haar wavelet is scaled down and its amplitude is scaled up by powers of 2 and $\sqrt{2}$ respectively.

Now, let's assume that we have an image I. A 2-D Discrete Haar transform of order $2^n \times 2^n$ (with $N=2^n$ ) goes as follows (P. Porwik, A. Lisowska 2004a):

$$\begin{cases} P = a \times H(n) \times I \times a \times H(n)^T \\ \quad where \\ H(n) \neq H(n)^T \quad and \;\; a = \dfrac{1}{N} or \dfrac{1}{\sqrt{N}} or \log_2(N) \end{cases} \tag{9}$$

P in formula (9) is a generated spectrum matrix by the product of the two Haar functions matrices H(n) and $H(n)^T$ with the Image I. This product is treated as an extractor of information (i.e. features) from an image I. The different generated coefficients of this product characterize particular image features as edges (P. Porwik, A. Lisowska 2004a) or other objects (D. Yankowitz, A. M. Bruckstein 1989).

## 2.2. Haar Like features

Correspondingly, Haar-like features are defined as Haar coefficients. Haar-like features were well detailed and described in the work proposed by Viola and Jones for real-time object detection such as face detection (P. Viola, M. Jones 2001a) (P. Viola, M. Jones 2001b). Some Haar rectangular filters, known also Haar-like filters, described in their work are illustrated in figure 11.

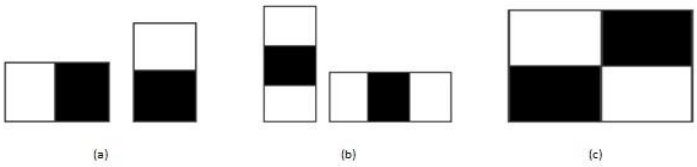


**Figure 11: Example of Haar rectangular filters used in object detection in the work of Viola and Jones (P. Viola, M. Jones 2001a).**

Each Haar rectangular filters is defined as a set of connected black and white rectangles. The Haar-like feature computed from applying a filter at an entry M (x,y) is obtained as follow:

$$H_F(M) = \sum_{Pixel_i \in Black_M} Pixel_i - \sum_{Pixel_j \in White_M} Pixel_j \qquad (10)$$

The difference between the sum of pixel intensities within white rectangle and pixel intensities within black rectangle defines the 2-rectangle feature (figure 11 (a)). The difference between the sum of pixel intensities of white rectangles and pixel intensities within the black rectangle represent the 3-rectangle feature (figure 11 (b)). Finally, the difference between diagonal pairs of rectangles represents the 4-rectangle feature (figure 11 (c)).

The variation of intensity between two adjacent regions may bring out information about the presence of edges or other specific objects. Yet, the application of Haar-rectangular filters on a given image allows locating specific contours.

The application of a Haar-like filter that is characterized by a Kernel (k) on an image I is a transform operation modeled by a convolution product and which it is defined as:

$$I_k(M) = \iint k(M - x)I(x)dx \qquad (11)$$

We are going now to show two obtained results by applying two Haar-like filters on the standard image Lena in the following figures (figure 12 and figure 13). We apply two filters (figure 11 (a)), where the sizes of both black and white rectangles are of 8×4 pixels, on the grey level Lena image. This permits generating global properties of the processed image.

In figure 12, we apply horizontal two-rectangle features. We obtain information about the horizontal lines in the image. Whereas, the transformed image, obtained by applying vertical two-rectangle features (figure 13), includes information about vertical lines in the image.

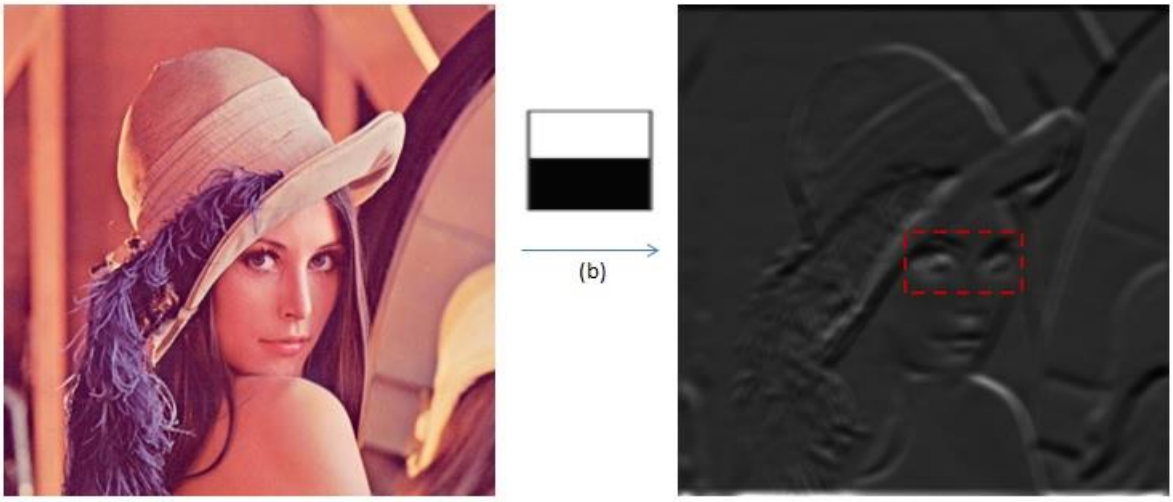


**Figure 12: Original image and the horizontal transformed image where the red box is the result of applying the filter on this specific region of the image.**

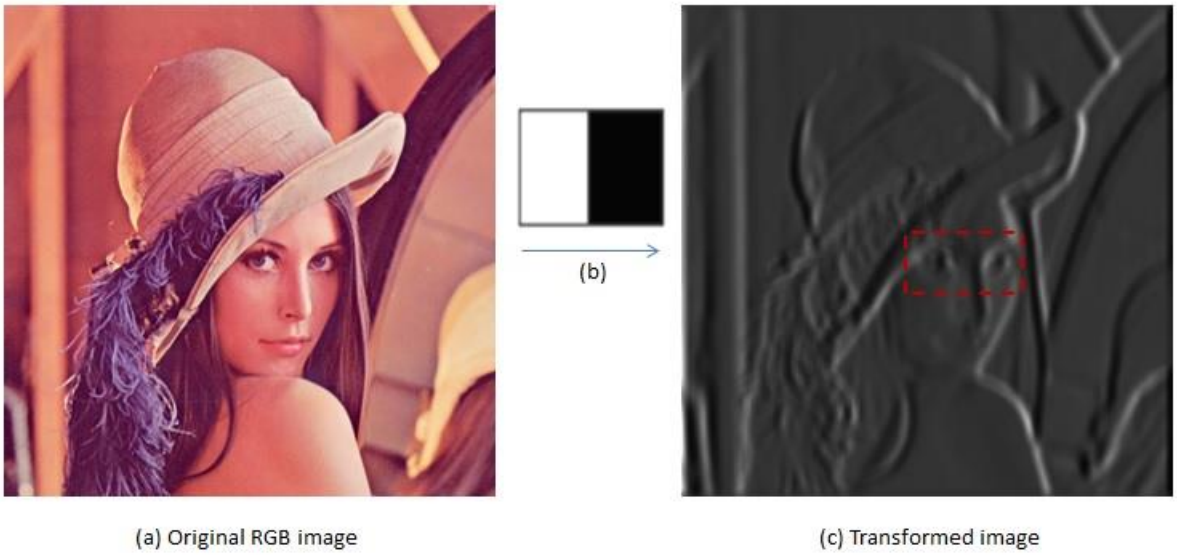


**Figure 13: Original image and the vertical transformed image.**

Furthermore, taking into consideration that the detector size used in the work of Viola and Jones is 24×24 (the size of the detector varies according to the application) and each K-rectangle feature (K=2, 3, 4) is applied at pixel level, so an intensive large set of rectangle features are generated (about 45,396 features). Besides, this computation is time and memory consuming. For instance, if we want to compute the sum of pixels within a small region 20×20, then we have to proceed with 400 memory accesses and 399 sums (formula 12):

$$Sum = \sum_{x=1}^{20} \sum_{y=1}^{20} I(x, y) \qquad (12)$$

However, if we increase the area of the region to be processed, the memory accesses and sum operations will also increment. For example, if the region is 100×100, then we will have a 10 000 memory accesses and 9999 sums.

Thus, for any region w×h, we will have w*h memory accesses with (w*h)-1 sums.

To overcome these constraints, Viola and Jones (P. Viola, M. Jones 2001a) proposed a new image representation called Integral Image (II) that permits computing very rapidly and effectively the Haar-like features.

# 3. INTEGRAL IMAGE REPRESENTATION

The Integral Image (II) representation is generated from a given greyscale image. The II is a greyscale image. Moreover, the integral image is mainly used in computing the average intensity within an image or computing the sum of pixel values within a template including connected rectangles.

The generation of the Integral Image representation goes in parallel with a generation of Summed Area Table (SAT) (F. Crow 1984). In the SAT, each input (x, y) has a value equal to the sum of the original pixel value of (x, y) with all the pixel values above and to the left of (x, y).

Formally:

$$II(x, y) = \sum_{\substack{x' \le x \\ y' \le y}} I(x', y') \qquad (13)$$

II is the integral image representation of the given image.

Correspondingly, the value of each pixel in the integral image II may then obtained as:

$$II(x, y) = II(x-1, y) + I(x, y) + II(x, y-1) - II(x-1, y-1) \qquad (14)$$

Otherwise, if we want to calculate the sum of pixels in some rectangle from an image (as it shown in figure 14), we use only four array references. This process is performed by the following equation:

$$I(x', y') = II(A) + II(C) - II(B) - II(D) \qquad (15)$$

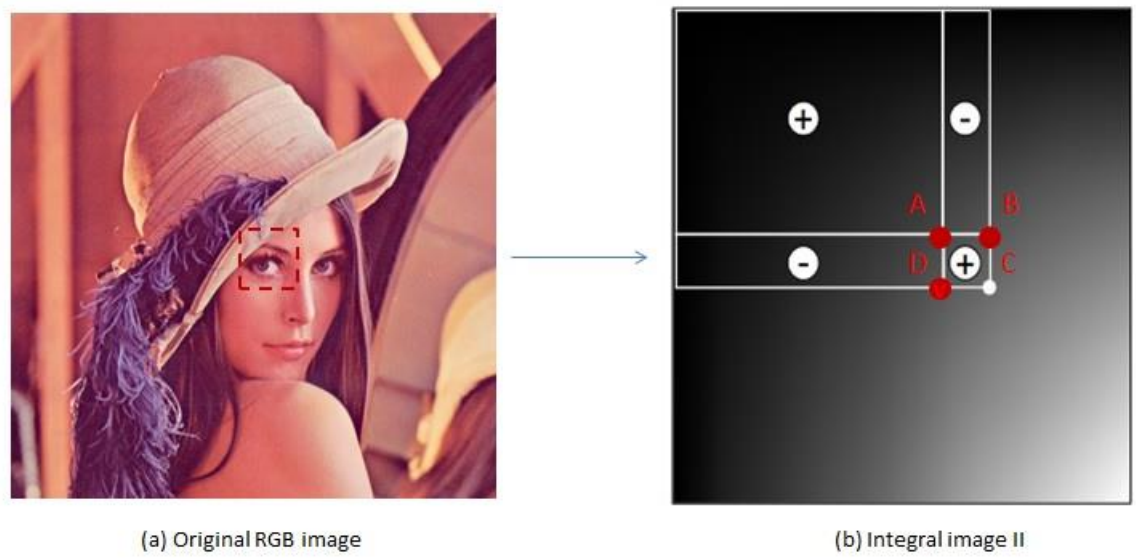


**Figure 14: An illustration of how to calculate a specified region (highlighted in red in (a)) an integral image II of an image I. This illustration highlights the equation 15.**

In this way, the performance of computing the overall sum of pixels in any rectangular region from the given image is done in constant time or in Θ (1) complexity. This is done independently of the size and position of the studied region.

Hence, the integral image II is an effective image representation that permits generating Haar-like features in constant time.

# 4. APPLICATIONS USING HAAR-LIKE FILTERS

Haar-like filters have been widely used in literature for object detection problems. We may organize Haar-like based approaches according to their topics.

## 4.1. Pedestrian detection

In this topic, the well-known approaches are proposed in (M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, T. Poggio 1997) and (M. Jones, P. Viola, D. Snow 2003). The first work for pedestrian detection has been introduced by (M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, T. Poggio 1997) in 1997. In this work, Papageorgiou et al. have presented a pedestrian detection approach using a Haar wavelet representation called the wavelet template. This approach allows detecting pedestrian within cluttered scenes.

The wavelet template is based on a subset of wavelet coefficients of the image in order to define the different shapes of objects within the image. This derivative of Haar wavelet is able to deal with highly non-rigid object with various sizes, shapes, texture, and colors. Thus, the main advantage of this wavelet is its ability to characterize complex objects and to be invariant to the variation of color

and texture. So, wavelet template is considered as one of the most efficient mechanism in object detection due to its invariant properties.

## 4.2. Face detection

Besides, in the face detection tasks, various works were proposed in this topic such as (R. S. stankovic, B. J. Falkowski 2003) (P. Viola, M. Jones 2001a) (P. Viola, M. Jones 2001b) (S. Du, N. Zheng, Q. You, Y. Wu, M. Yuan, J. Wu 2006) (M. Jones, P. Viola 2003).

The most well-known works are proposed since 2001 by Viola and Jones (P. Viola, M. Jones 2001a) (P. Viola, M. Jones 2001b) that based their face detection approach on Haar-like features and an AdaBoost learning algorithm in order to select the efficient critical visual features that help in efficient classifiers (Y. Freund, R. E. Schapire 1995). Those classifiers are combined in a cascade fashion to classify quickly the foreground objects such as faces and the background objects. This approach achieves high face detection accuracy despite there exist various sets of conditions such as illuminations, scale, and camera variations. However, this approach is sensitive to rotated objects.

Another work in this topic was proposed in (R. Lienhart, J. Maydt 2002). The contribution of this work is the introduction of an extended set of Haar-like features that are twisted at 45° (R. Lienhart, R. Kuranov, V. Pisarevsky 2003). They were able to overcome the major drawback of the work of Viola and Jones. For that, they have proposed a novel technique to compute the Integral Image for 45° rotated features. However, those rotated features are computed in a two-pass process using four references.

Subsequently, the authors in (A. L. C. Barczak 2005) and (C. H. Messom, A. L. C. Barczak 2009) have extended the set of Haar-like features. By using additional Integral Images, rotated object at different angles such as 11.3°, 26.5°, 63.4°, and 78,6° can be accurately detected. Figure 15 shows some example of k-rectangle features.

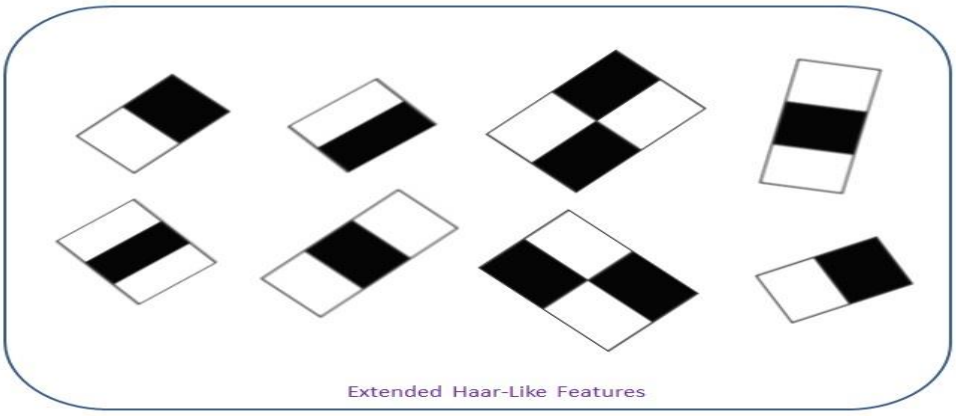


**Figure 15: Samples of extended Haar-like features in distinct angles. Those extended features may detect rotated objects at different angles.**

We conclude from above that Haar-like features can be generated whatever the object angles. However, these features have symmetry properties meaning that the positive and negative areas in the feature have to be equal areas. To overcome this constraint, (G. A. Ramirez, O. Fuentes 2008) have used a set of asymmetric Haar-like features for face detection.

### 4.3. Other object detection

Here, we are going to enumerate various approaches that are based on Haar-like features in the literature. We highlight that the use of Haar-like features in these works is done as a preliminary step before a learning phase. Face tracking and classification approaches using Haar-like features are introduced in (M. Jones, P. Viola 2003) (J. Barreto, P. Menezes, J. Dias 2004) (R. Lienhart, L. Liang, A. Kuranov 2003).

Vehicle detection problems are studied in (P. Negri, X. Clady, S. M. Hanif, L. Prevost 2008) (A. Khammari, E. Lacroix, F. Nashashibi, C. Laurgeau 2005). Other approaches have been introduced in (A. L. C. Barczak , F. Dadgostar , C. H. Messom 2005) (M. Kolsch, M. Turk 2004).

For head pose detection and tracking task, adaptive Haar-like features were used in (N. J. Pyun, H. Sayah, N. Vincent 2014) . Moreover, we found other approaches for other real time nature task as robot soccer ball detection (S. Mitri, K. Pervlz, H. Surmann, A. Nchter 2004) and wild life surveillance (T. Burghardt, B. Thomas, P. J. Barham, J. Calic 2004).

## 5. CONCLUSION

In this chapter, we described Haar functions and Haar-like features and we enumerate some image processing and pattern recognition approaches based on them. As we adopt only the use of Haar-like filters in our context, so the conclusion is going to be essentially for Haar-like filter.

Indeed, Haar-like features have been widely used in object detection problems. First, only simple symmetric horizontal, vertical, and diagonal Haar-like features were computed from an Integral image representation of the given image. Then, lots of extended works have been proposed to enlarge the ability of detecting object at different angles and various sizes of detectors. Moreover, we conclude that Haar-like features have lots of properties such as:

- Computationally efficient due to the integral image presentation: they are quickly computed from the integral image.
- Computed at distinct angles.
- Local oriented intensity differences.
- Sensitive to the presence of edges and other simple image structure.
- Support effective learning.
- Allows real time detection.

Aside these properties, Haar-like features for object detection are considered as considerable features to be adopted because they are simple, not memory consuming, and computationally efficient.

*"If the facts do not fit the theory, change the facts*

**"_Albert Einstein**

# IV. Text and graphic separation in comics

## Contents

# 1. MOTIVATION

Entertainment and storytelling may be considered as an art form. They represent an artistic work full of imagination and creativity. However, the types and forms of stories vary among the different communities. Generally, the well-known forms of storytelling are as follows: Text, Film/Tv, Theatre, Legend, Oral Traditions, Fable, Cave Paintings, and Myth. One of the typical forms for storytelling for children, adolescents or even adults is comics. Comics have been existing with different representations in human culture (S. McCloud 1994).

Therefore, there is a lack of accessibility to comics because most often they are materialized as books, magazines or even daily papers. But, taking into consideration the technological advances nowadays, comics are proposed to be downloaded on digital devices such as mobile devices (K. Arai, H. Tolle 2011) (Cyb 2009) (Y. In, T. Oie, M. Higuchi, S. Kawasaki, A. Koike, H. Murakami 2011). Thus, they are widely involved in our living and especially our leisure times.

This expansion has encouraged the community of document and graphics researchers to analyze the large amount of comic albums. So, Comics analysis domain is considered as a new research area full of challenging problems with different constraints. Accordingly, different applications have been recently proposed for graphic documents such as indexation, the search for specific items, and content analysis (e.g. text localization, speech balloons detection, and frame detection). Nonetheless, few studies have been done in this research area. This fact has encouraged us to move our researches towards such type of document and to propose some original techniques to overcome some existing constraints in the literature.

One crucial step in the direction of fully automatic comic books understanding is text localization. Thus, we focus in this chapter on the text/graphics separation problem where text detection and localization is essential.

We give in the next section an overview about the nature of comic documents and the different methods proposed in the literature for text localization.

# 2. STATE OF THE ART

Comic books can be known under different names. This is based on cultural particularities of each country and its spoken language (C. Ponsard, V. Fries 2008). Besides, comics are considered as complex graphical documents. Their background is of a graphical nature. The textual elements can be randomly situated and oriented with various fonts, style, alignment, size, and colors. Besides, there are various content types within each comic document. Thus, the nature of this type of documents is different from the classical structured documents.

Despite all of that, comic documents share almost the same design. Each page contains a sequence of frames or strips which are separated by white or mono-color tubes or gutters. More simply, a strip may be defined as a sequence of drawings arranged in interrelated panels. Indeed, we may also find other contents such as:

- Texts written in a short narrative.
- Speech balloons or bubbles that usually tend to be round or square shapes with a tail pointing to a present comic character.
- Action words that are written in capitals within colored jagged splats.
- Graphical pictures.

These contents can be either contained within each frame or overlap two frames or more (A. K. Ngo Ho, J. C.Burie, J. M.Ogier 2011). More particularly, the text may be written within speech balloons or on the outside of them (see Figure 16).


**Figure 16: Some samples of the written text location in comics. (a) An RGB comic image where text is written in black within speech balloons. (b) Two RGB images where text is written outside speech balloons. The written text may be darker than the background color or brighter than it.**

As a matter of fact, most of state-of-the-art techniques that treat text detection and localization tasks in comic books based their works on assumptions which are "text is part of speech balloons" and "text is written in black in a white speech balloon background" (K. Arai, H. Tolle 2011) (M. Yamada, R. Budiarto, M. Endo, S. Miyazaki 2004) (K. Arai, H. Tolle 2010a) (K. Arai, H. Tolle 2010b).

Hence, frame and speech balloons detection steps must to be achieved before text extraction step. The authors of (M. Yamada, R. Budiarto, M. Endo, S. Miyazaki 2004) propose a sorting rule to extract all text areas within the different speech balloons after gathering them.

Furthermore, in order not to restrict the application by the assumption "text is written in black in a white speech balloon", (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011) propose an application for frame and text extraction from comic pages where text background color should be similar to page background.

Moreover, there are two different categories in the literature for text/graphic separation and text localization in comic documents which are described in sub- section 2.1 and sub-section 2.2.

## 2.1. Top-down approaches

The first work in this category is introduced by (M. Yamada, R. Budiarto, M. Endo, S. Miyazaki 2004). The authors proposed to define a sliding window in order to detect characters. The size of the sliding window is nearly the same size as the different characters present in the processed comic page. Then, another work in this category has been recently introduced by (K. Arai, H. Tolle 2011) where the authors used various mathematical morphology operations to detect lines. The flow diagram of this process is presented in figure 17. The limitation of this work is that only closed speech balloons are detected.

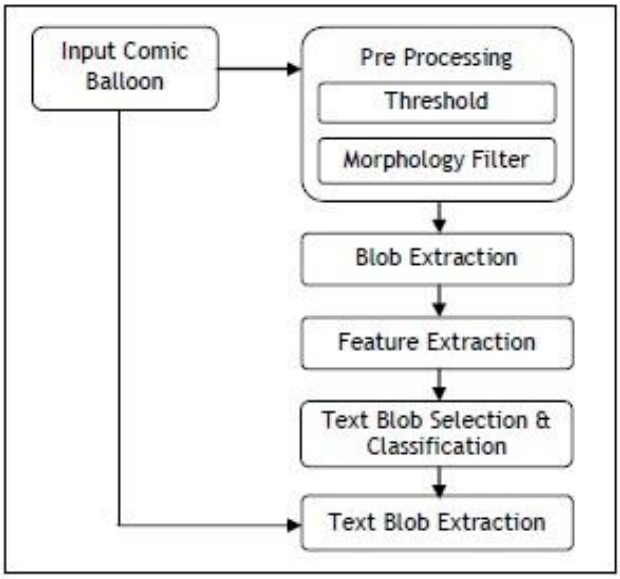


**Figure 17: the text detection and extraction process in (K. Arai, H. Tolle 2011).**

More precisely, the different components of the text extraction process are as follows: (1) input comic image. (2) Frame extraction. (3) Balloon detection. (4) Text blob extraction. (5) Text recognition/OCR. (6) Extracted text. Hence, the text extraction step may be done only after detecting balloon speech within the extracted frames.

## 2.2. Bottom-up approaches

Recently, few numbers of works have been proposed (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011) (C. Y. Su, R. I. Chang, and J. C. Liu 2011) (C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, J. M. Ogier 2013). Approaches introduced by these works are based on the Connected Component Algorithm (CC). A segmentation step is primordial. In (C. Y. Su, R. I. Chang, and J. C. Liu 2011), the proposed algorithm allows classifying text from non-text CC by using an SVM classifier. This needs a learning step that specifies the field of the application of the method. For the text/graphic separation process, sliding windows are used. The limitation of this approach is that it is orientation and resolution dependent. Then, each processed comic element may be classified into 3 classes: “Text”, “Noise”, or “Frame”. The classification process is based on CC heights (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011). Despite of the simplicity of the classification process, it works only when the pro-

cessed page contains text with a background color similar to the paper background. This is because this approach relies on a binarization process before detecting text areas, and this binarization step assumes that the text background brightness is similar to page background. As a continuity of the latest described work, (C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, and J. M. Ogier 2013) has proposed a text/graphic separation technique based on local contrast ratio. Their approach is size, translation, rotation, scale and contrast invariant. However, it fails in cases where there are graphical texts as page title and bright text over dark background. Besides, it has some difficulty in detecting certain types of text such as graphic sounds. This approach allows detecting more than 75.8% of text lines with 76% precision.

Furthermore, we may also think of text detection in graphics, architectural plane or others. The complexity of the graphic in comics is even higher but the number of text words is generally larger. Then, some specific methods are needed.

In the context of our work, we propose an automatic text extraction method for digitized comics. This method is not based on any of the previous assumptions. Nevertheless, we only assume that text has to be either nearly horizontal or nearly vertical (e.g. the Japanese writing).

# 3. TEXT AND GRAPHIC SEPARATION

In this section, we are going to describe in details our proposed approach of text detection and localization in comics. The flowchart of the framework is presented in figure 18.

## 3.1. Text detection in comics

To detect text areas from processed comics (figure 18), the idea is to apply Haar-like features that are easily computed from the Integral Image as described in previous sections (Chapter III section 3). As far as we know, this is the first work that applies Haar-like features on digitized comics.

**The question is**: Why applying filters as Haar-like filters on comic pages will bring out accurate results in text/graphic separation process?

**The response is**: Text, in most cases, introduces in the document some horizontal or vertical well-contrasted parts, whatever the colors of the text or the background are. Then, looking at the page, text brings some horizontal or vertical contours. Thus, we estimate from the previous chapter (Chapter III) that Haar filters are very recommended to detect text areas in comics.

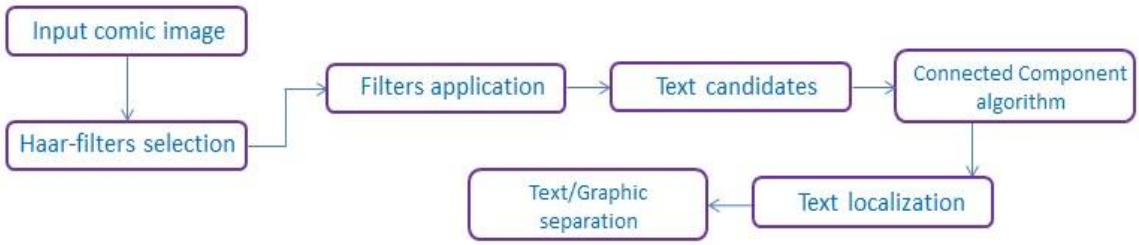


**Figure 18: The proposed approach for text and graphic separation process. This diagram enumerates the different steps of our proposed framework. The application of generalized Haar-like filters on the input comic image results on various zones of interests or candidates. The Connected component analysis permits separating text from graphic components.**

In the next two sub-sections, we are going to define the used GHFs for text extraction in our work and describe the process of text and graphic separation in comics.

### 3.1.1. Generalized Haar filters for text detection in comics

Haar-like features permit detecting contour zones of writing text in document images. In fact, Generalized Haar-like Feature (GHF) are recently proposed and applied to heterogeneous document collections to detect queried texts (A. Ghorbel, J. M.Ogier, N. Vincent 2015).

The fundamental principle of this technique is based on applying globally a certain number of viewpoints which are Haar-like filters on the processed comic images. These viewpoints will respond to the upper and lower parts of the text lines. In fact, they correspond to a global transformation of the comic image that allows highlighting different shapes. These viewpoints are independently treated and their findings are gathered together to make the final results.

Indeed, the applied filters must be constructed taking into account one major characteristic which is the size of the writing along the processed comics. But, this size has not to be estimated in a precise way as only the position of text lines has to be detected. Then, after defining the filters that are characterized by a Kernel (K), we apply globally each kernel. In fact, the application of the GHFs generates several Haar-like features. These features, which indicate certain characteristics in the image (e.g. presence of edges, changes in texture) are positioned in the middle of the line separating the white rectangle from the dark one.

For comics, we define some asymmetric Haar-like rectangles filters as it is shown in figure 19 (b). Practically, the number and the size of the applied GHFs may vary depending on the characteristics of the text within the processed comics. The black rectangles correspond to the writing part and the white rectangles corresponds to non-writing part. Each one of the defined filters is considered as a 2-rectangle feature that indicates where the border lies between a light and a black region. These filters will allow detecting aligned text areas, either the top part of text (figure 20 (d-1)) or the bottom part of text (figure 20 (d-2)). We assume that the height of the black rectangles is larger than the white one in order to get accurate responses by applying the GHFs. Also, other elements rather than the text may respond to the filters.

Moreover, the determination of the height of the black and white rectangles should be done taking into account the size of the text. The height of each pattern is expected to almost fit the text line height. Within most of comic pages, readable texts are noticed to be written in small sizes with respect to other components heights (e.g. panels, speech balloons, comic characters). However, they are large

enough to be easily seen by the reader. As we choose asymmetric filters and taking into consideration the characteristics of the processed comic pages, we set the height of each filter to 21 pixels, where the height of the black rectangle is considered to be the two-thirds of the filter height. This is chosen in order to be coherent with the space between two successive text lines and then to highlight more text areas.

### 3.1.2. Generalized Haar filters application for text detection in comics

Indeed, applying an asymmetric GHF generates a grey transformed image ($I_k$) (figure 19(c)) representing the input comic image. This application process is a transform operation modeled by a convolution product and which it is defined as:

$$I_k(p) = \iint K(p-x)Com(x)dx \qquad (16)$$

This process helps us to highlight several regions that represent candidate text zones in each processed comic image (Com). Generally, the number of convolution process depends on the number of the GHFs applied. In our context, the convolution operations are performed for each filter at each pixel of the processed comic.

Then, we sum up all information from the different obtained transformed images into one grey image, known as the accumulated image ($I_{accum}$). The process of information accumulation is performed by a superposition of obtained transformed images. The accumulated image highlights all candidate zones (figure 21 (b))

. The Formula modeling this step is as follows:

$$I_{accum} = \sum_{k \in \square} I_k \qquad (17)$$

When we focus on each transformed grey image, we observe potential responses within it. These responses indicate a possible presence of a text line region. Accordingly, these responses are more representative in the accumulated image as they reinforce the existing of text candidate zones. Thus, we are going to apply a binarization step in order to highlight the results.

The process of binarizing the transformed accumulated image is defined by the formula 18.

$$I_{cw} = Bin(I_{accum}) \qquad (18)$$

$I_{cw}$ represents the resulting binary image. After this step, we convert the binary image to a color image where we assign the red color to the candidate zones (figure 21 (c)). Figure 21 (d) illustrates more clearly the red candidate pixels.

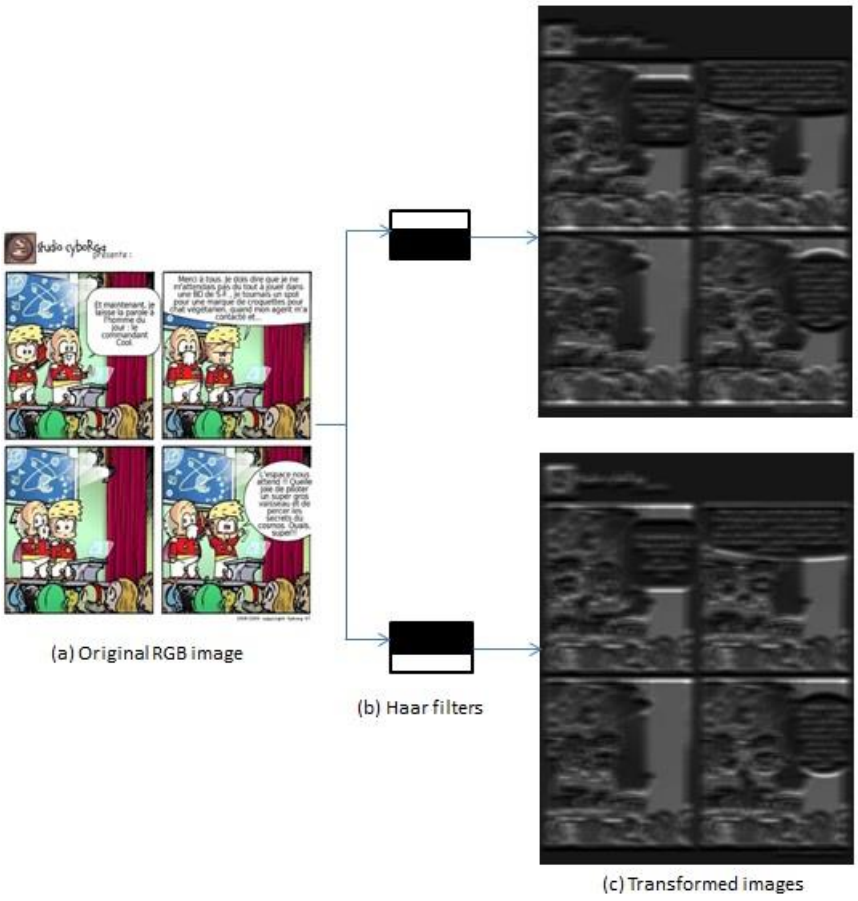


**Figure 19: The process of applying asymmetric Haar-like features in digitized comics. (a) The RGB input comic image where text is written in black within white speech balloons and outside them. (b) The two asymmetric Haar-like applied filters. The first one in the top highlights the top part of the texts while the second one highlights the down part of the texts. (c) Two grey transformed images resulting from applying the two asymmetric GHFs. In both transformed images, candidate text zones and other non-text element present a high answer, highlighted in white in both transformed images, regarding the nature of the applied GHF.**

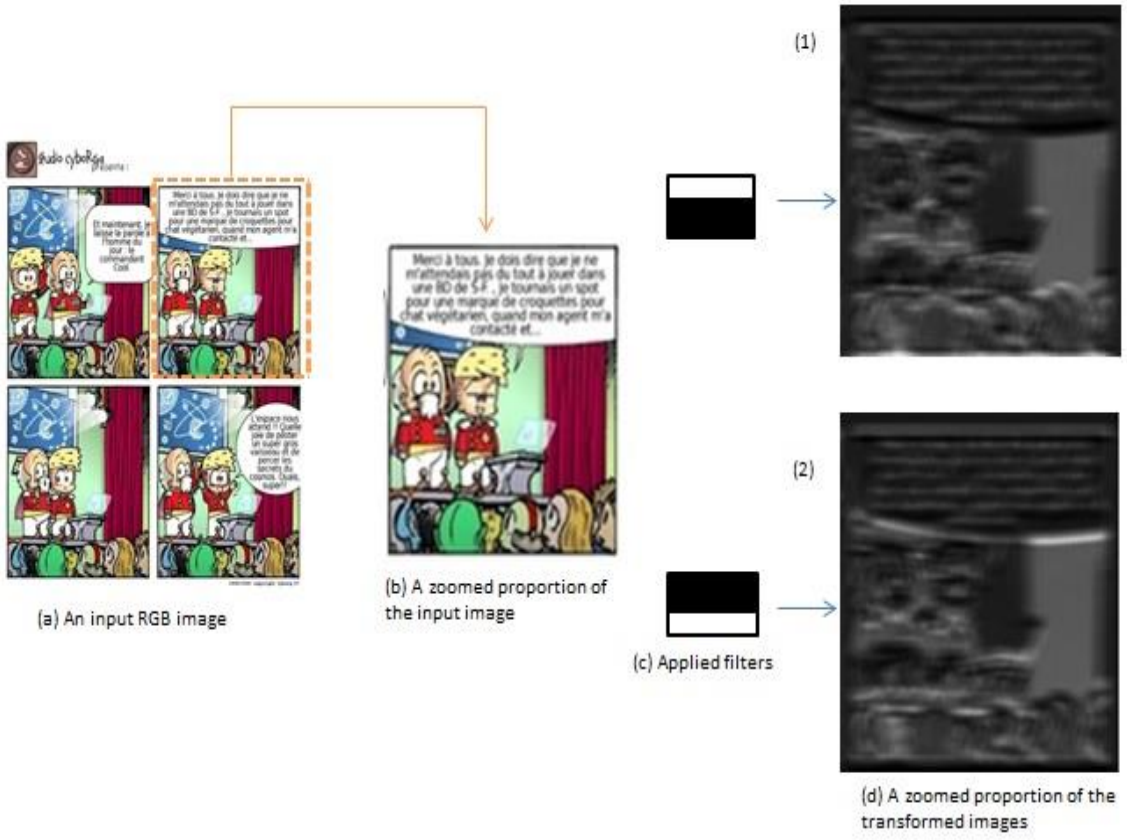


**Figure 20: A zoomed illustration of applying asymmetric filters to an RGB comic image. (a) An RGB input comic image. (b) A panel containing comic characters, speech balloon, and a text written in black within white speech balloon. (c) The two asymmetric Haar-like applied filters. (d) Two grey zoomed portions of the transformed images resulting from applying the two asymmetric GHFs. (d-1) the first proportion highlighting the top part of the text. (d-2) the second proportion highlighting the down part of the text.**


**Figure 21: The text detection process based on asymmetric Haar-like features in digitized comics. (a) The RGB input comic image where text is written in black within speech balloons with various background colors. (b) The accumulated grey image resulting from applying two asymmetric Haar-like filters (Figure 19(b)) on the input comic. (c) The result image where text areas are represented by horizontal red shapes. (d) A zoomed portion of the final image.**

Thus, our method permits detecting text lines within comic images whatever their colors or their positions in the image. Nevertheless, it also allows detecting graphical elements as it is shown in figure 21 (c) for instance. Thus, an automatic technique to specify only texts components is recommended. This technique will grant a separation between texts and graphic elements in the processed comic.

## 3.2. Text and graphic separation technique

The detection stage results in generating many candidate zones that generate either text components or graphical elements. The idea is to connect detected components sharing some characteristics into groups that will discriminate words or text line regions from graphical components.

In order to connect detected regions into larger components, we use the Connected Components (CC) labeling algorithms. This latter allows grouping all the connected pixels in an image into blobs or components based on their connectivity (R. Szeliski. 2010). We applied the CC algorithm on each

binary image generated by the text detection technique (section 3.1). The connected blobs (Figure 22 (b)) are figured thanks to their bounding boxes.

Indeed, this algorithm will rely on the following assumption:

- Each detected pixel is assigned to a connected component.

After generating the different blobs, we classify them into two categories:

- Blobs constituting candidate representing textual elements.
- Blobs constituting candidate representing non-textual elements.

The criterion that allows assuming if connected components represent textual elements or not is the aspect ratio between the width and the height. In fact, texts in comics are most of the time smaller than other graphical element such as panels, speech balloons, etc. Bounding boxes representing textual elements should have a height value smaller than the width value. The criterion is based on this assumption. The value representing the median of all bounding boxes areas enable us to retain the bounding boxes associated with candidate text regions. Area values that are less than the median value are retained as candidate text region, and they will be deleted otherwise. As a consequence, we are able to localize the text within the processed comic (figure 22 (c) and figure 23 (c)). After this filtering pass, we may obtain several bounding boxes that are not associated with textual elements due to their sizes (figure 22 (b)). Therefore, we perform another filtering pass by eliminating bounding boxes with too small areas.

The next two figures (figure 22 and figure 23) illustrate the process of separating text candidate regions from graphic in comics.

(a) Original RGB image

(b) Candidate text regions detection with Bounding Boxes

(c) Text localization

**Figure 22: Text detection and localization result. (a) The RGB comic image where the text is written in black inside and outside speech balloons with white and yellow background color. (b) Bounding boxes associated with candidate text regions. (c) Text separated from graphic elements of the input image. Connected components with small bounding box areas are eliminated. The color map of the text is randomly taken.**

(a) Original RGB image

(b) Candidate components detection

(c) Text localization

**Figure 23: Text detection and localization result. (a) The RGB comic image where the text is written in black inside and outside speech balloons with different background color. (b) Text and graphical components are detected after applying the CC algorithm. (c) Text separated from graphic elements of the input image. Connected component algorithm is applied in order to localize text regions. The color map of the text is randomly taken.**

In this chapter, we have described two steps of an approach that permit detecting and localizing text within comics. We are now going to evaluate this approach in the next section.

# 4. EVALUATION

In this section, we describe first the evaluation protocol and metrics adopted in our work to evaluate the generated experiments.

## 4.1. Dataset description

The proposed text detection and localization approach has been evaluated on the eBDtheque database (C. Guerin, C. Rigaud, A. Mercier, F. Ammar-Boudjelal, K. Bertet, A. Bouju, J. C. Burie, G. Louis, J. M.Ogier, A. Revel 2013). We mention that there is no other dataset publicly available for the evaluations rather than this dataset. Indeed, it contains 100 comic book images with the newest ground truth version known as "version 2014"[7]. This database is characterized by the diversity of formats, and styles of comics. Furthermore, comic pages are different in design and printing techniques as they were published in the mid-twentieth and 21st Centuries.

The database is consisting of:

- 46% of images scanned from French comic books at 300 DPI (Dots Per Inch) in A4 format
- 37% of images scanned from French web comics with various definitions and formats
- 11% of public domain American comics[8] scanned at 300 DPI in A4 format
- 6% of unpublished artwork of Japanese manga scanned at 72 DPI in A5 format
- 5% of the images contains a double page
- 29% of the images were published in the mid-twentieth Century
- 71% of the images were published in 21st century

The comic pages are composed of:

- 850 panels
- 1550 comic characters
- 1092 balloons
- 4691 text lines

As the objective of our work is text detection and localization in comics, we are now going to give more information about text through this database.

Most of comic pages within this database contain text written in French and only 13 and 6 comic pages contains respectively English and Japanese text. Besides, the majority of text is upper-case and it is printed or handwritten. We differentiate two types of text: speech text and narrative text. Text in both types is usually written within balloons.

Indeed, onomatopoeia or more precisely graphic sounds, which are considered as a particularity of comics, are presented within 18 pages. Despite of speech and narrative text, onomatopoeia do not require any container to be written in. In our context, onomatopoeia is considered as a text part. Furthermore, punctuation symbols (i.e. question mark or exclamation mark) and illustrative text (i.e. road sign) are assumed to be as text line as well.

[7] http://ebdtheque.univ-lr.fr/
[8] http://digitalcomicmuseum.com/

## 4.2.Evaluation metrics

We evaluate our approach for text extraction in terms of object bounding boxes. Evaluation is similar to the PASCAL VOC challenge (S. Fidler, J. Yao, and R. Urtasun 2012) and the work proposed by Rigaud (C. Rigaud, C. Guérin, D. Karatzas, J. C. Burie, and J. M. Ogier 2015).

Indeed, the evaluation of the proposed approach is done by using different thresholds N. Each threshold value N represents the overlap minimum percentage between regions to be detected and detected regions. α is the overlapping percentage between the predicted text and its complete corresponding text in the ground truth (formula 19). Yet, the evaluation is done at text line level. The computation of the different evaluation metrics is related to the used threshold.

$$\alpha = \frac{area(B_t \cap B_{gt})}{area(B_t \cup B_{gt})} \tag{19}$$

Where $B_t$ is assigned to the bounding box of the predicted text (i.e. the detected text) and $B_{gt}$ represents the bounding box of the corresponding text in the ground truth.

Moreover, the Recall (R) and Precision (P) rates are computed by using the number of TP, FP, and FN (false negative or missed elements). After that, we compute the F1 score that is the harmonic mean of recall and precision where its best value is 1 and the worst is 0.

These rates are modeled by the following three formulas:

$$R = \frac{TP}{TP + FN} \tag{20}$$

$$P = \frac{TP}{TP + FP} \tag{21}$$

$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{22}$$

Now, after describing the evaluation protocol and metrics used for the evaluation, we are going to show our experimental results qualitatively and quantitatively.

## 4.3. Experiments

### 4.3.1. Qualitative results

To show that our proposed approach is able to overcome some existing limitations proposed in the literature (section 2), we will introduce in this sub-section some qualitative results.

First of all, we are able to detect **all text zones within speech balloons whatever the color of texts and the background of the balloons are**.

The following two figures (figure 24 and figure 25) show some examples of detecting black text written in white speech balloon and examples of detecting black text in a non-white speech balloon regions, respectively.

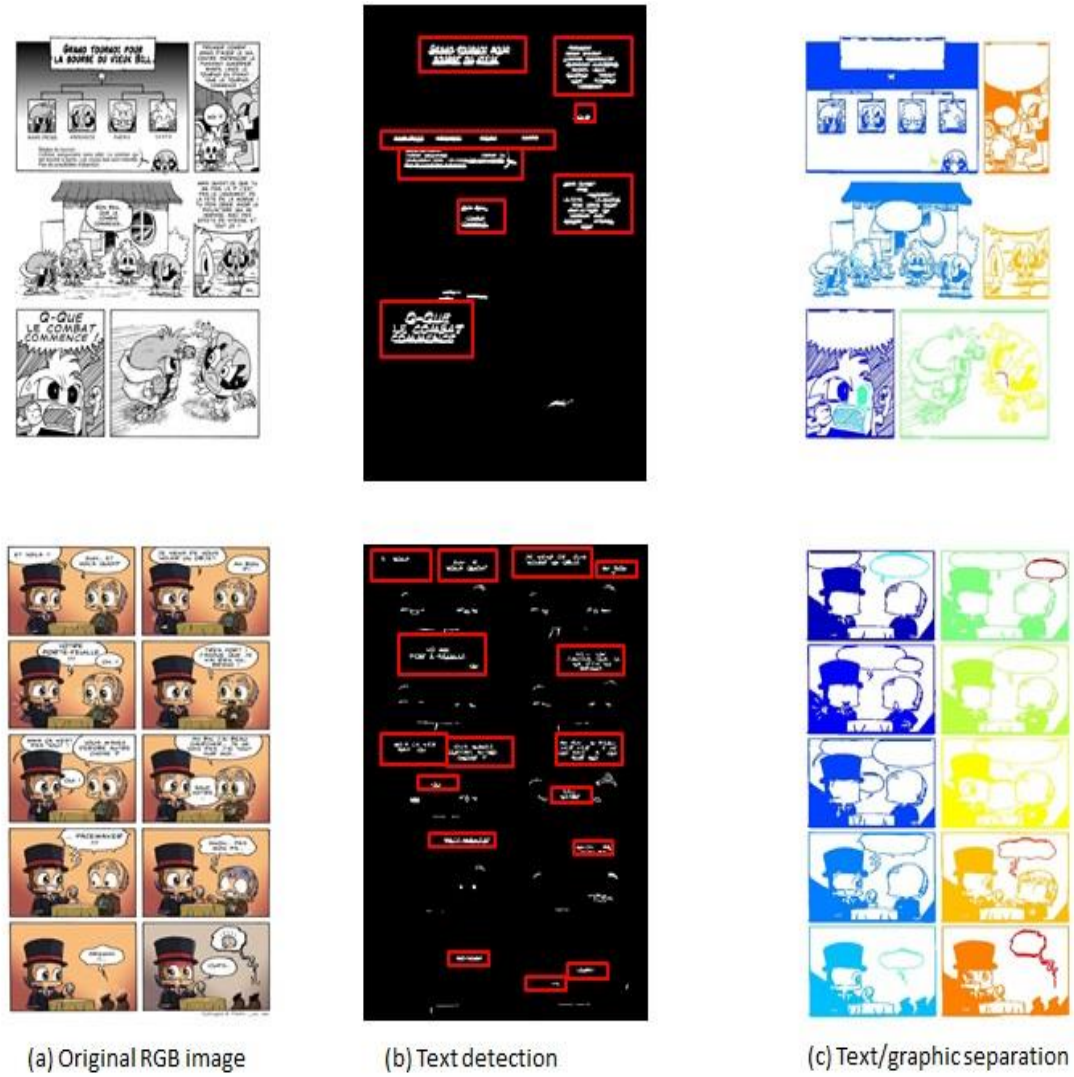


**Figure 24: Text/graphic separation in the case of black text written in speech balloon with white background. (a) RGB processed comics. (b) Text detection results. The red rectangles illustrate detected texts. (c) Final results of text extraction from graphics**

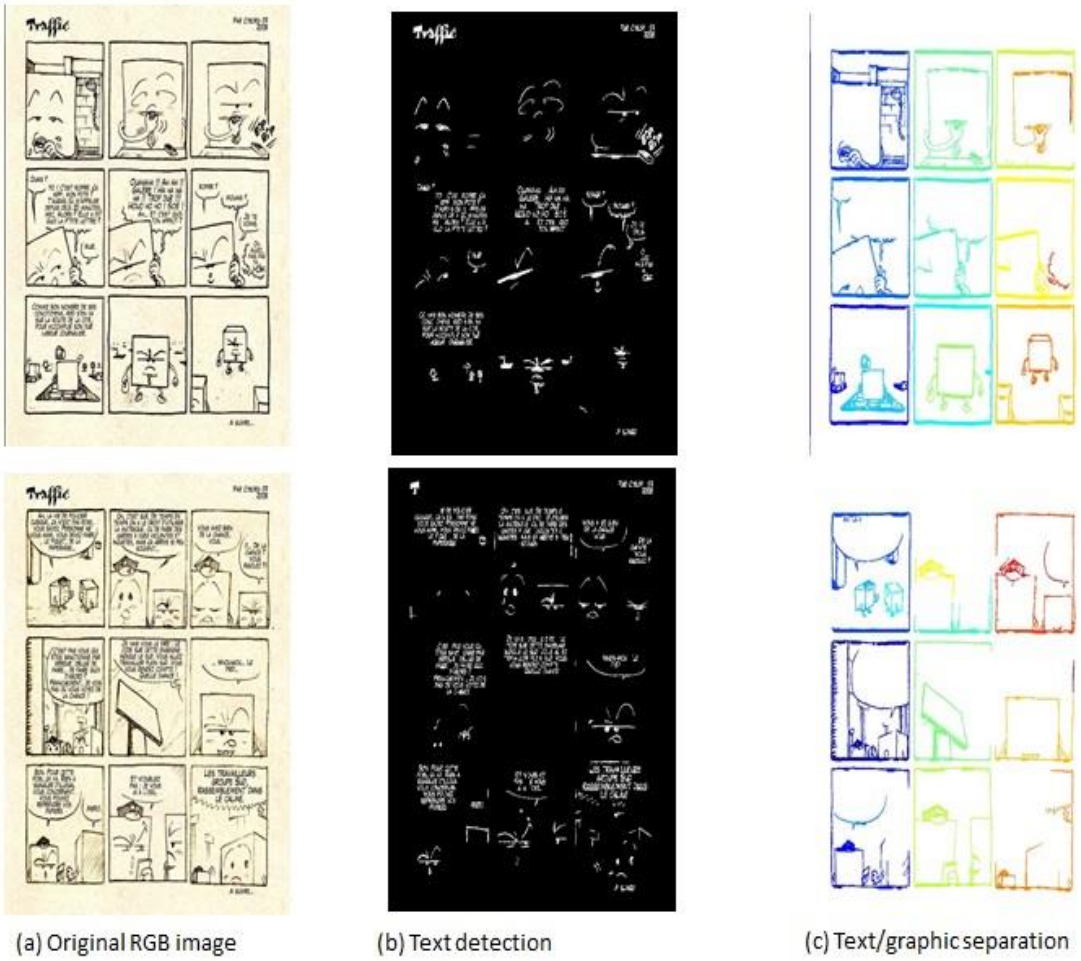


**Figure 25: Text/graphic separation in the case of black text written in a non- white speech balloon. (a) RGB processed comics. (b) Text detection results. Some false positives are highlighted. (c) Final results of text extraction from graphics. Graphical components are restored.**

Secondly, the assumption **"text background color should be similar to page background"** is overwhelmed in our work.

To illustrate that, we show some obtained results in the following figure (figure 26).

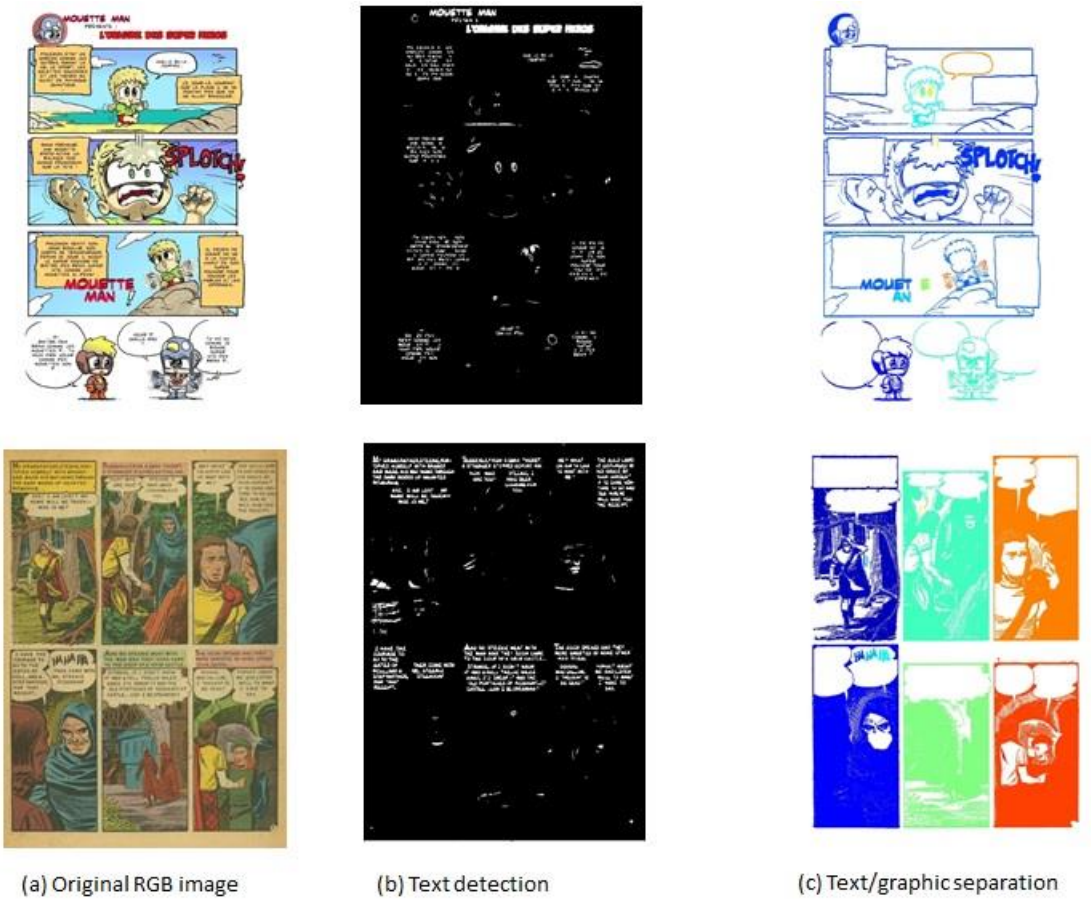


**Figure 26: Text/graphic separation where text background color is not similar to page background. (a) Original RGB comic images. (b) Text areas and FP are highlighted. (c) The graphical components are separated from the detected texts.**

The Figure 26 shows that our proposed approach is not restricted to Assumption **"text background color should be similar to page background"**. Indeed, the background color of the text is different from the color of the background of the comic processed page (Figure 26 (a), (b)). In figure 26 (a), there are: texts written in black within white balloons, texts written in black within yellow balloons, and the text background color is not the same with the page background color. Figure 26 (c) shows that some text zones have not been detected. These false negatives correspond to curved texts.

Thirdly, the assumption "**text is part of speech balloons**" does not present a constraint in detecting and localizing texts all over comic images.

Indeed, localize text lines outside speech balloons represents a real challenge for spotting text lines from comic pages. We illustrate in figure 27 some generated results by our proposed approach where there are some texts that are not written within speech balloons.

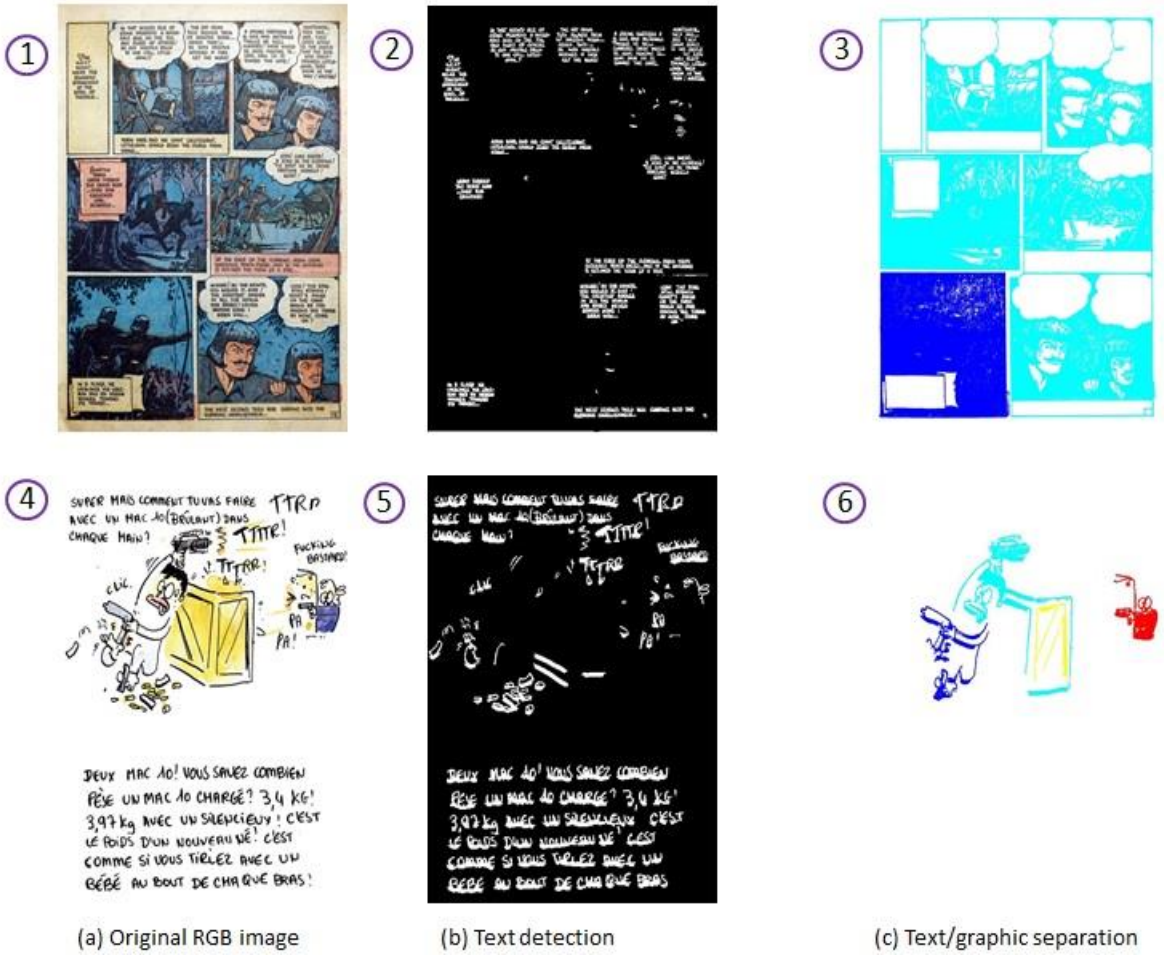


**Figure 27: Text/graphic separation where text can be written outside the speech balloons. (a) RGB comic images where text are written inside and outside the speech balloons. (b) Text areas and FP are highlighted. (c) The graphical components are separated from the detected texts.**

From figure 27, we bring out two observations. First, figure 27 (c-3) illustrates that the process of text separation from graphic components within comics is done in accurate way whatever the location of the text (inside or outside speech balloons). Second, in figure 27 (a-4), text lines are not perfectly horizontal and they are written directly in the page. However, No speech balloon occurs. Despite of that, text lines are well detected (figure 27 (b-5)) and localized (figure 27 (c-6)).

Furthermore, our proposed approach can effectively detect punctuation symbols such as question mark or exclamation mark in comic images. The following figure highlights this conclusion (figure 28).

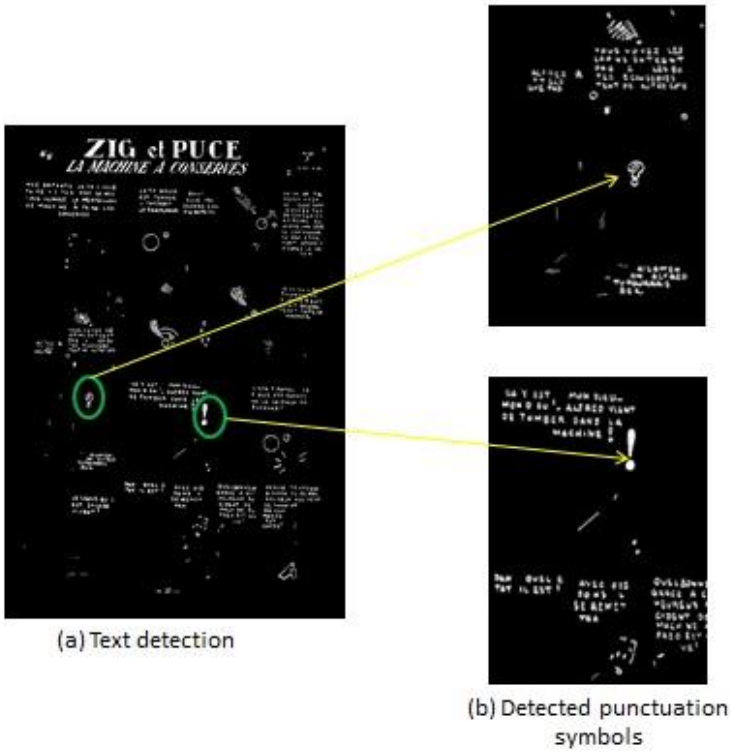


**Figure 28: Various types of punctuations detected in comics.**

Finally, we conclude from the qualitative evaluation several pros & cons.

The proposed framework does not rely on assumptions about the localization or about the written color and style of the text in the processed comics. Indeed, the whole process is done globally on the processed comic image without applying frame or speech balloons detection techniques. Another strong point is that this method allows detecting text inside or outside the speech balloons and generating accurate results.

Nonetheless, our work fails where there are highly non-straight text lines curvature texts in the comic page and when the text color is brighter than the background color. For the case where we deal with curved text, the use of rotated generalized Haar filters may lead to overcome this limitation. This is considered as a future work. Besides, when text is written in white within a black background, we should reverse the property of rectangles defining the applied filter. Thus, the white rectangle should represent the writing and the black rectangles should represent the non-writing. By applying such filters, we will generate candidate zones where we have writing brighter than background color. Figure 29 shows some examples of these cases.

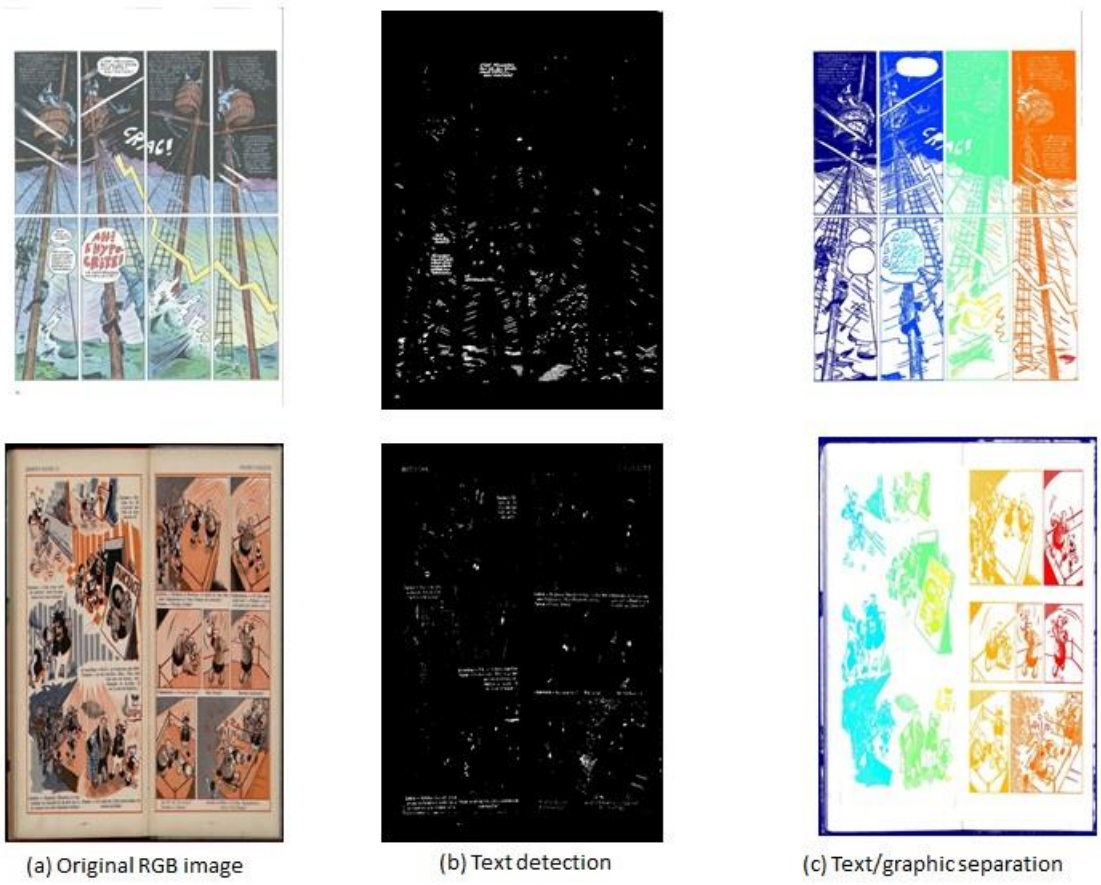


**Figure 29: Some examples illustrating two failure cases: (i) the case of curved texts, (ii) the case of text color brighter than background color (a) Original RGB comics where we may find curved text and some text lines having color brighter than the background color. (b) Text areas and FP are highlighted. (c) The graphical components are separated from the detected texts.**

### 4.3.2. Quantitative results

The quantitative results are obtained by evaluating our approach for text detection and localization on the 4691 text lines of the eBDtheque database "version 2014"[9] in terms of object bounding boxes at text line level. All the pages in the database are processed in the evaluation process.

[9] http://ebdtheque.univ-lr.fr/

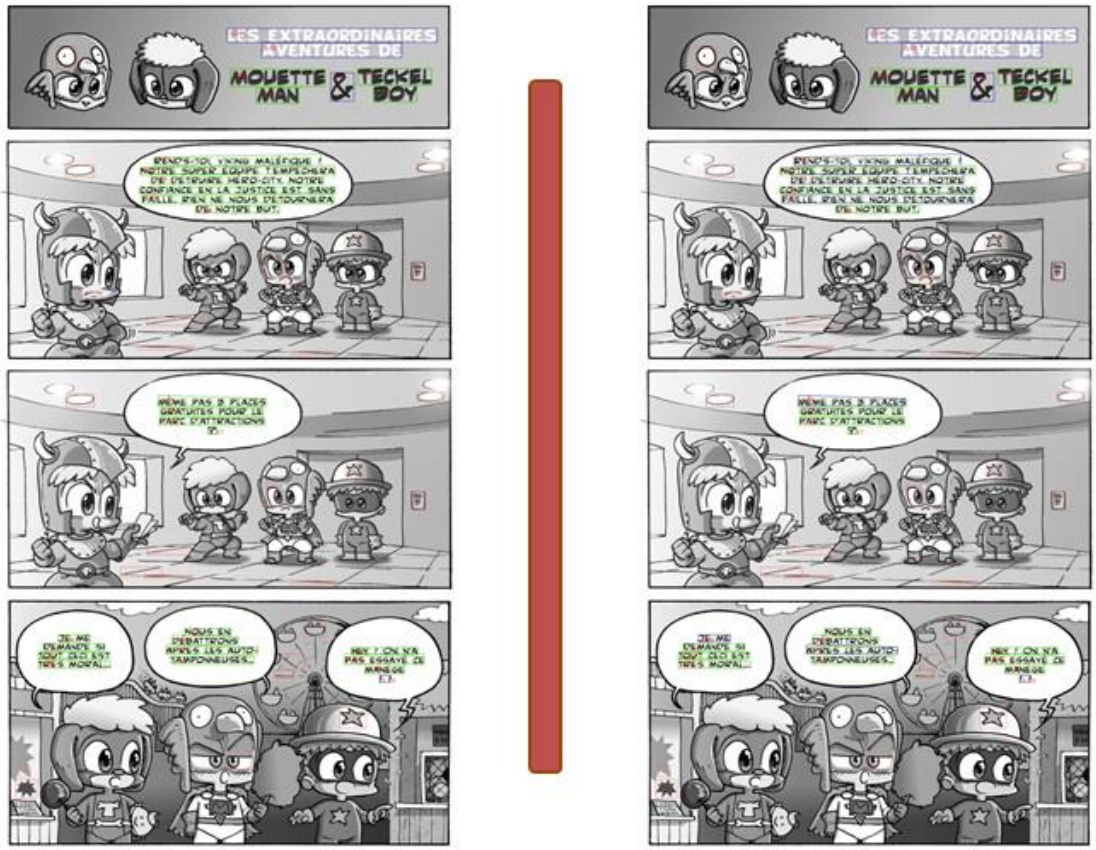


**Figure 30: Different results for different threshold values. (a) A threshold value of 50%. (b) A threshold value of 80%.**

In order to well understand this evaluation process, we present two obtained results with threshold values of 50% and 80% respectively (figure 30). Before describing the results, we are going to indicate the meaning of the different colors in both images (figure30 (a) and figure 30 (b)).

- Light green: bounding boxes of regions detected automatically
- Orange: false positives
- Red numbers: overlapping percentage of the couple (detected, ground truth)
- Dark green: validated regions
- Dark blue: invalidated regions

In the figure 30 (a), the dark green color dominates the image. That illustrates that using a threshold of 50%, a lot of text regions are well detected. However, increasing the threshold value to 80% (figure 30 (b)), some text areas that were detected with the threshold value of 50% are now not detected. The overlap ratio between detected texts and the corresponding text in the ground truth varies by varying the threshold value N.

Furthermore, we notice the existence of few orange rectangles. These rectangles correspond to false positives.

Other examples of text detection results are shown on the following figure (figure 31).


**Figure 31: Different results for a threshold value of 30%.**

During the qualitative evaluation, we have observed the existence of false positives (figure 24-figure 28). In order to decrease the number of the false positives, the idea is to perform a post-processing step after the text localization step. The post-treatment step is based on a Connected Components (CC) analysis technique. In this filtering step, we take into consideration areas of bounding boxes of localized text. Bounding boxes areas that are smaller than the average area of the overall bounding boxes areas of localized text are deleted. However, the deleted elements may represent textual elements and may not. In such case, the question goes as follows: the use of a post treatment step will increase or decrease the recall and precision scores, even if its use will delete various non-textual components?

To answer this question, we made a comparison study between the detection and localization of text lines results with and without performing a post-treatment filtering step.

Yet, we are going to introduce the recall and precision representing the text extraction of various images in the database obtained by varying threshold values N in two different cases:

- No post-treatment technique is performed.
- A post-treatment technique is performed.

#### 4.3.2.1. Text extraction scores with no post-processing.

We compute the Recall and Precision values of line detection by varying the threshold values by varying the overlap threshold value N.

The next 5 figures show the text extraction scores for each image with the threshold values of 10%, 30%, 50%, 70%, and 90% respectively. The overall evaluation is presented in figure 37.

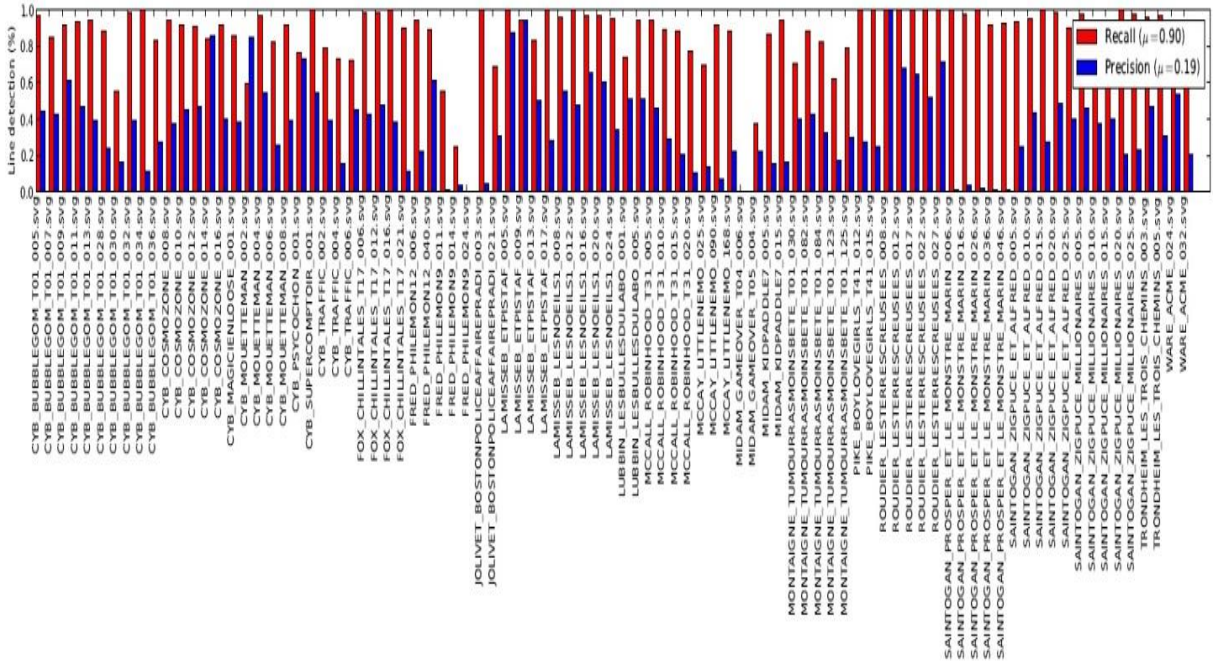


**Figure 32: Text extraction score details for different images of the database with N=10%.**

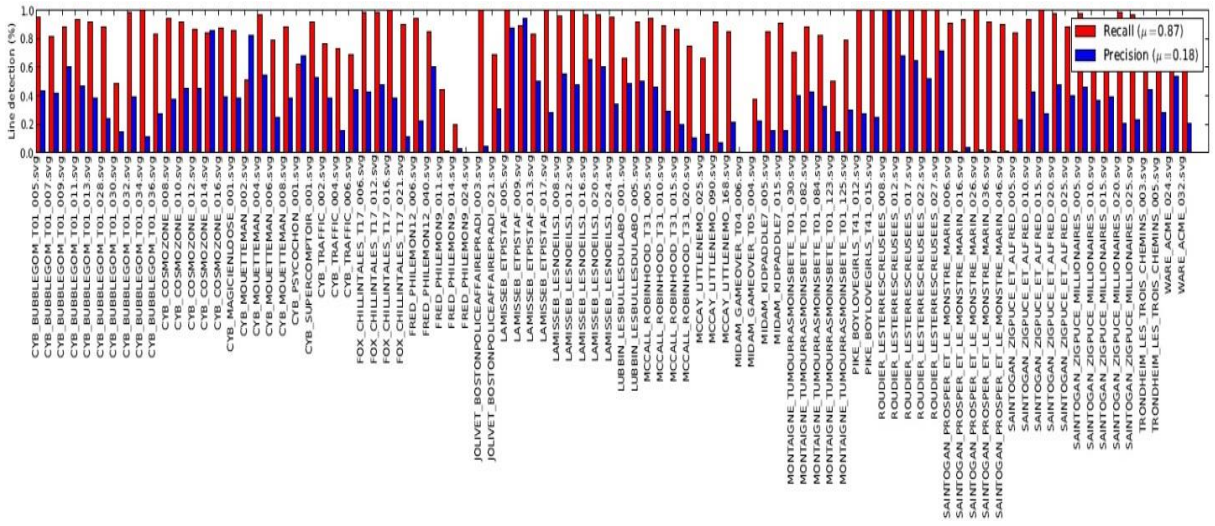


**Figure 33: Text extraction score details for different images of the database with N=30%. To this overlap threshold value, the recall and precision rates are almost stable as those obtained with N=10%.**

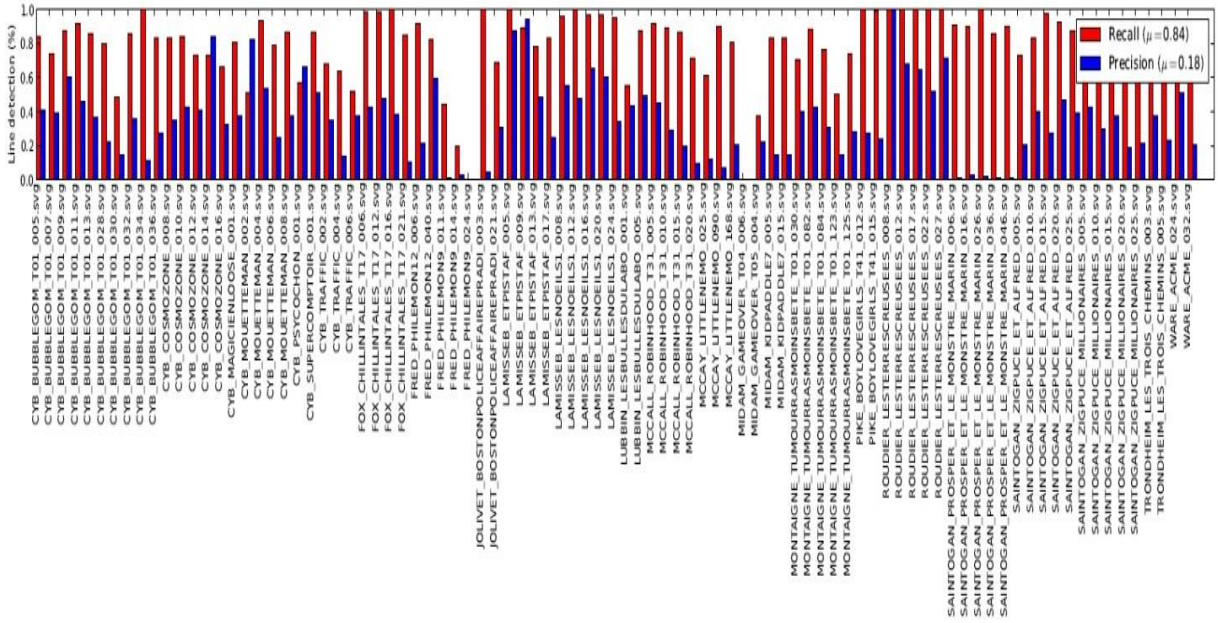


**Figure 34: Text extraction score details for different images of the database with N=50%.**

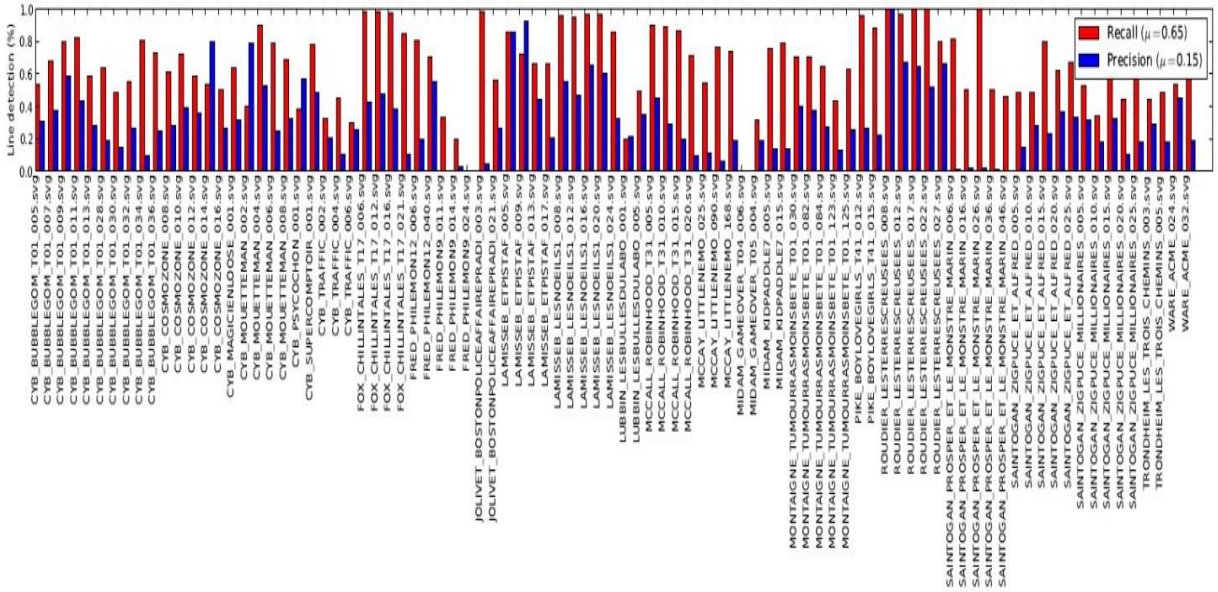


**Figure 35: Text extraction score details for different images of the database with N=70 %.**

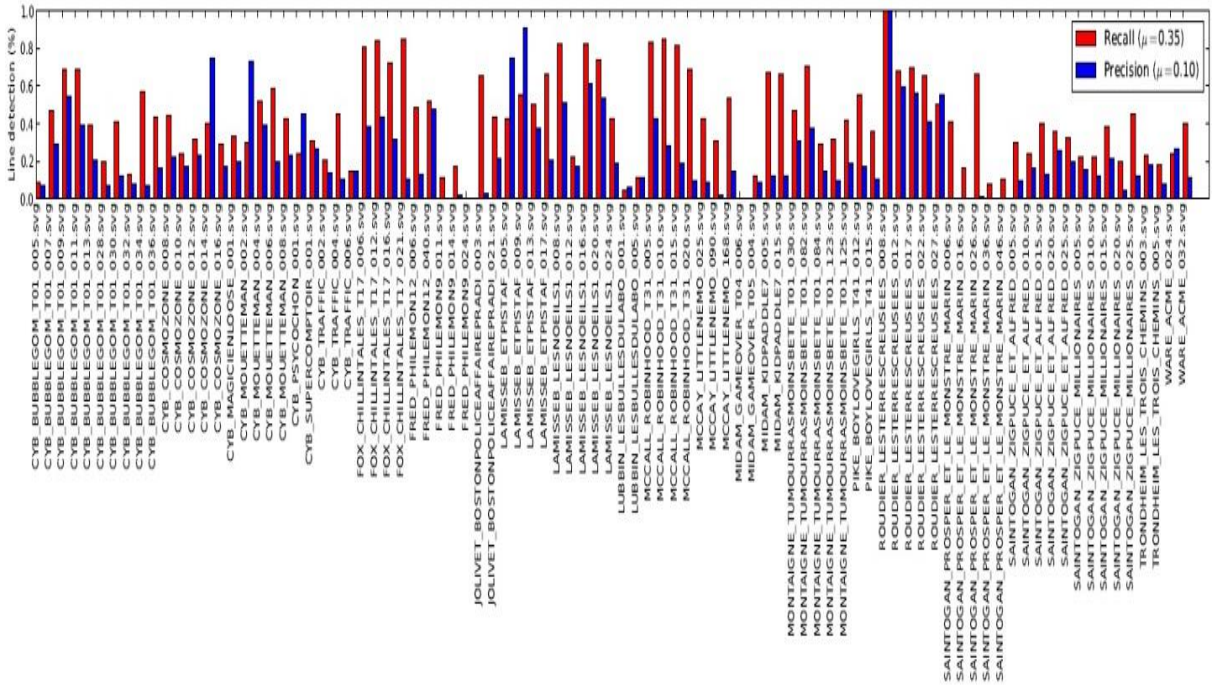


**Figure 36: Text extraction score details for different images of the database with N=90 %. After the overlap threshold value goes over 50%, the variation of both recall and precision rates becomes noticeable.**

In figure 37, the graph shows that the optimum values of N that gives the best recall and precision rate (around R=90% and P=20%) is 30%. On the one hand, the recall values are decreasing softly between the threshold values of 5% and 50%. However, it abruptly decreases since N is larger than 50%. This is due to the fact that the detected false negatives influence the recall rate. On the other hand, the precision rate is not so good compared to the recall rates due to the large number of false positives.

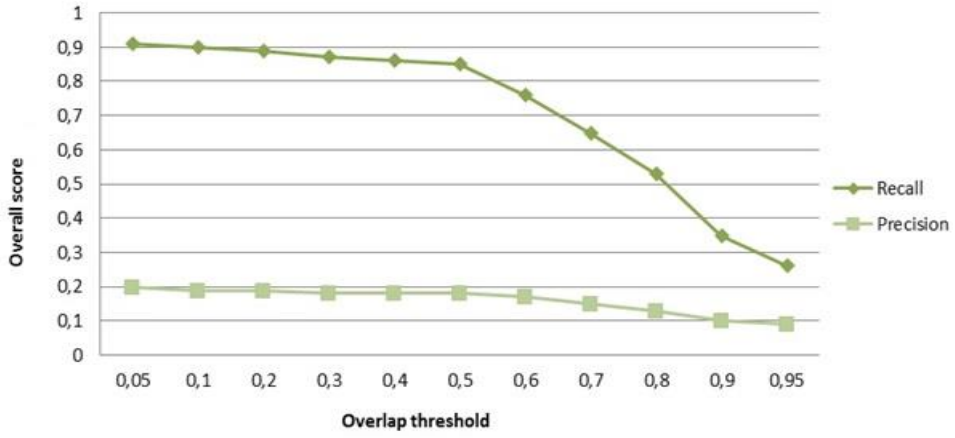


**Figure 37: The overall evaluation of text extraction without a filtering process. We tested the overall evaluation with overlap threshold varying from 0.05 to 0.95.**

#### 4.3.2.2. Text extraction scores with post-processing.

Forasmuch as the recall rate can reach 90% but the precision rate is low with almost of 20% when N is varying from 5% to 30%.

Yet, after performing the filtering step by applying the CC filtering technique, we illustrate in the next figures the new text extraction scores for each image with the threshold values of 20%, 30%, 35%, 40%, 45%, and 50%. Then, the new overall evaluation is presented in Figure 44.

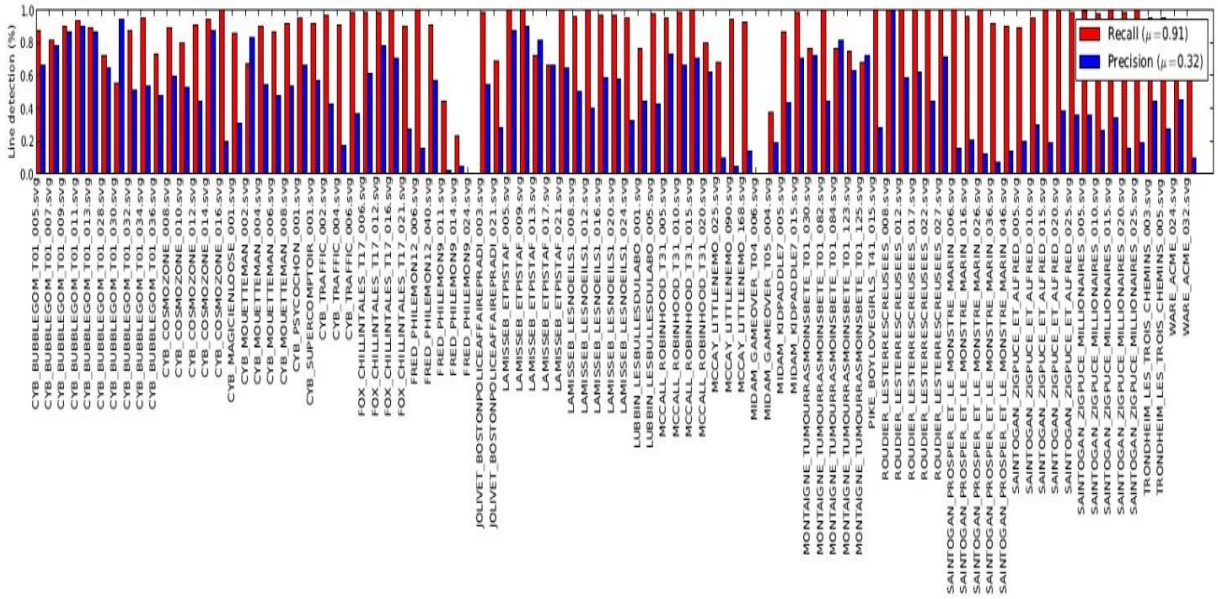


**Figure 38: Text extraction score details for different images of the database with N=20%.**

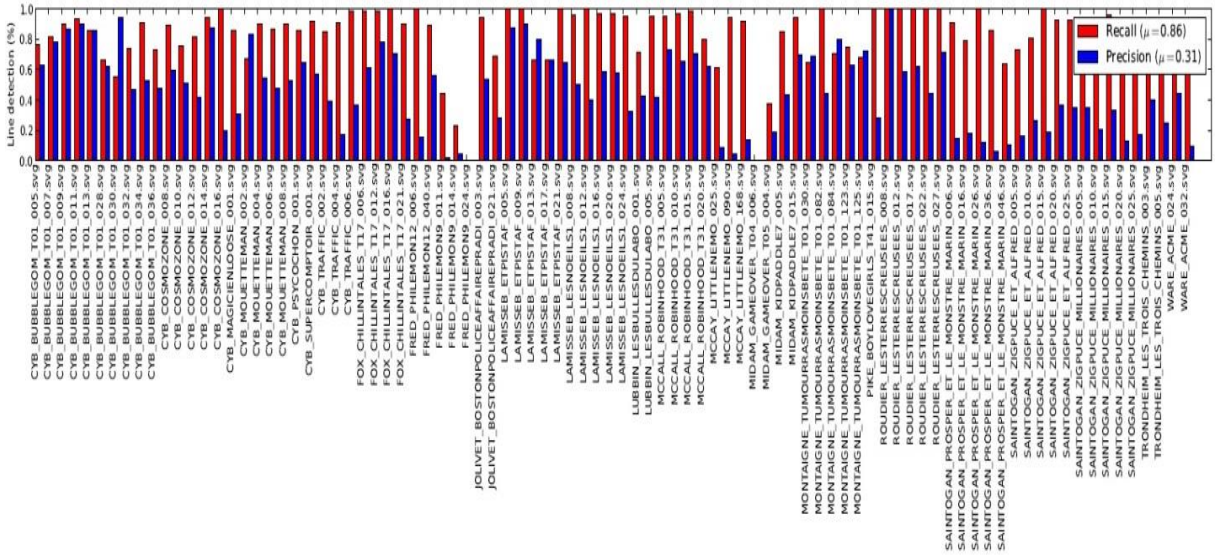


**Figure 39: Text extraction score details for different images of the database with N=30%.**

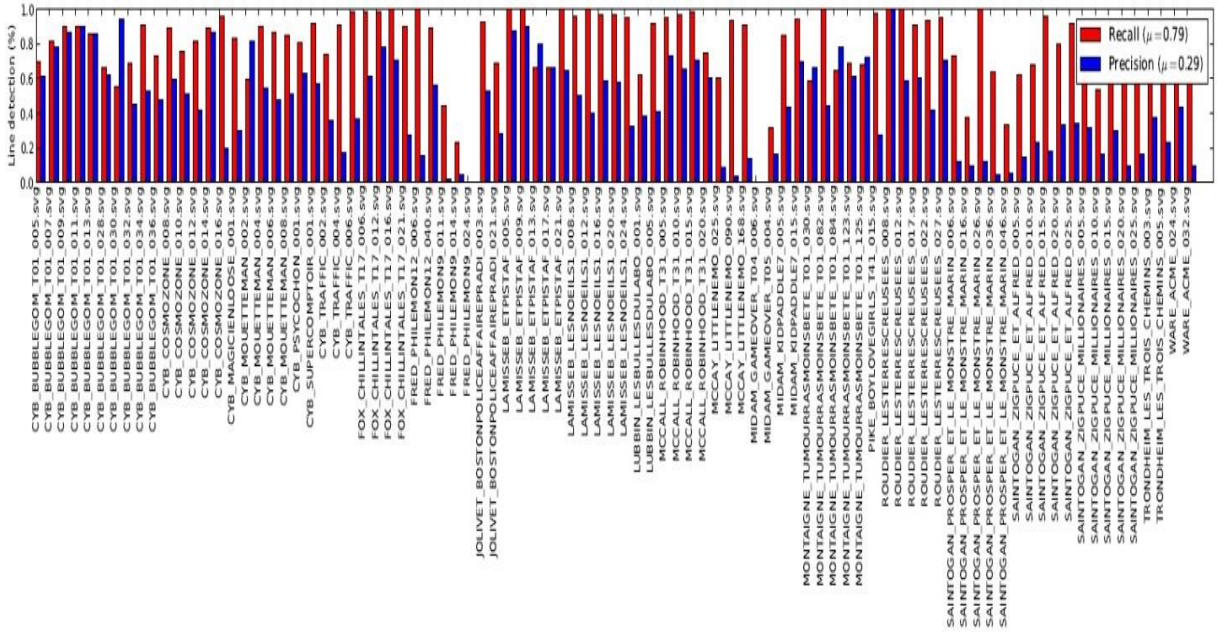


**Figure 40: Text extraction score details for different images of the database with N=35 %.**

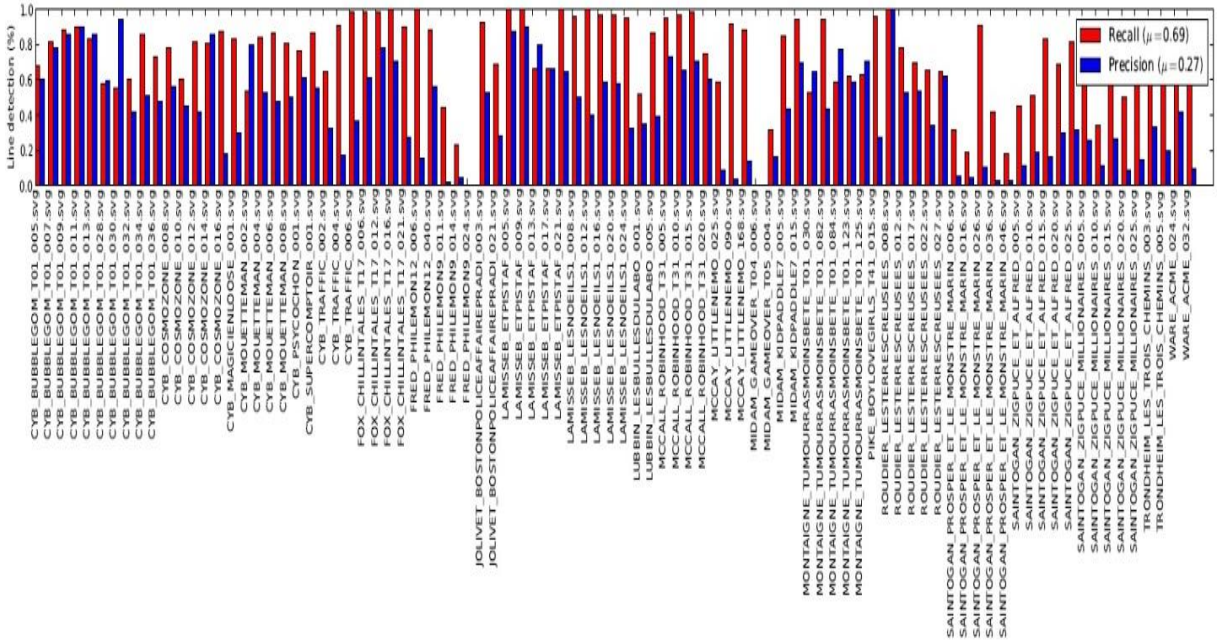


**Figure 41: Text extraction score details for different images of the database with N=40%.**

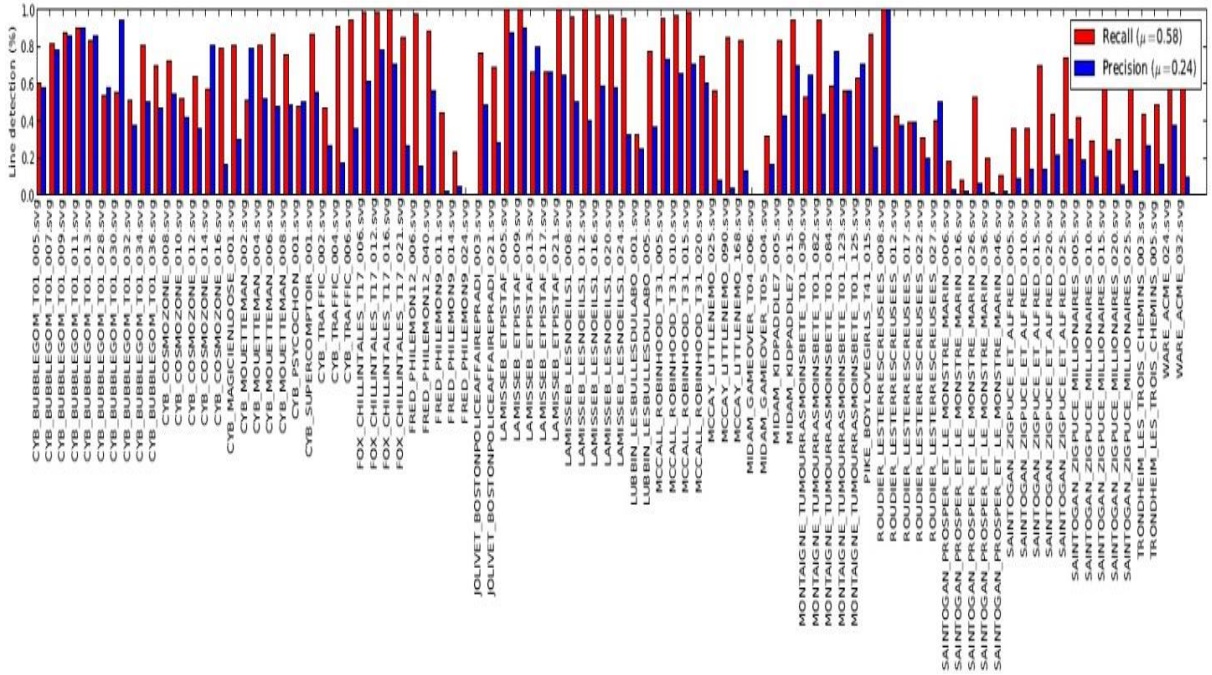


**Figure 42: Text extraction score details for different images of the database with N=45%.**

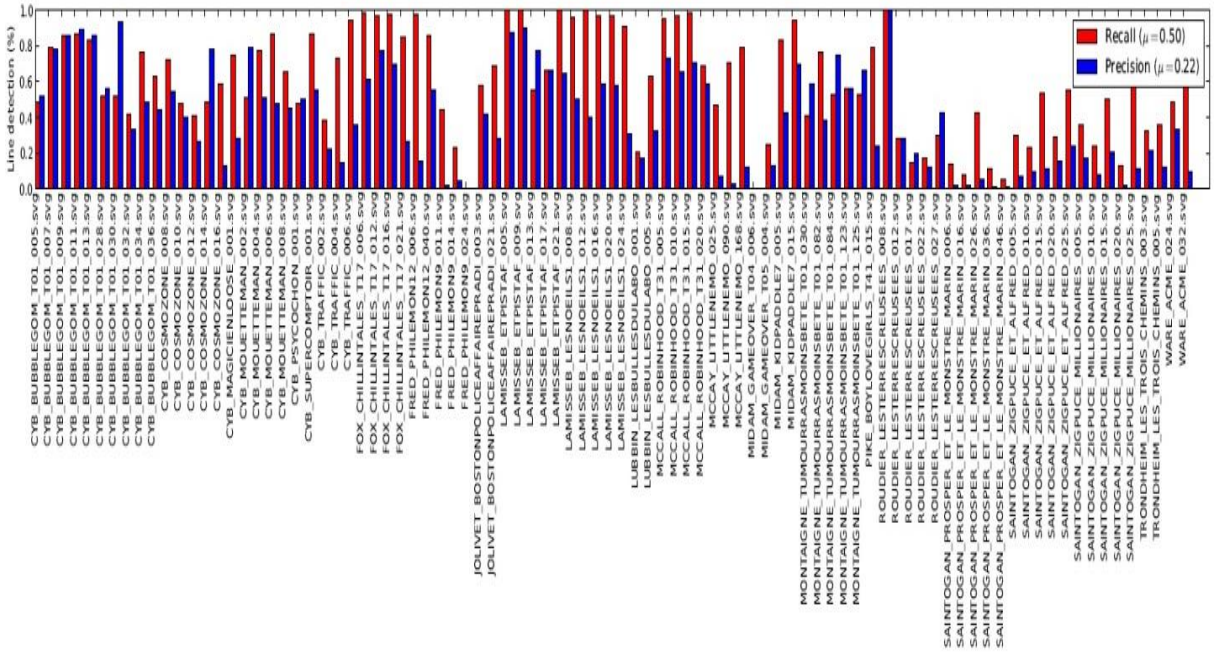


**Figure 43: Text extraction score details for different images of the database with N=50 %.**

In figure 44, the graph shows that the optimum values of N that give the best recall and precision rate (around R=93% and P=32%) are lower than N=30%. This could result in significant errors in the text recognition (OCR). Our approach is able to locate a large part of text areas, max recall >90%, even though they are sometimes only partial (with a min recovery at 5%). Yet, the average recall rate is very good. However, the maximum precision rate is over 30% with an overlap threshold between [0.05, 0.3].

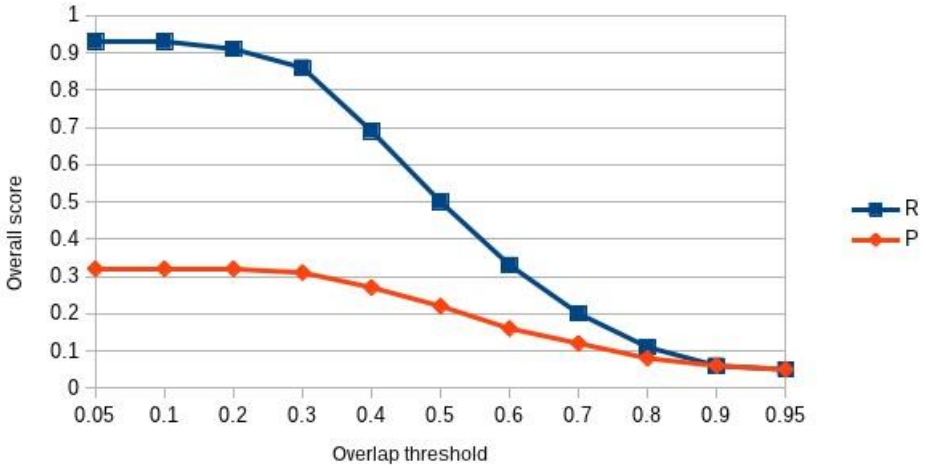


**Figure 44: The overall evaluation of text extraction with a filtering process. We tested the overall evaluation with overlap threshold belonging to [0.05, 0.95].**

To see clearly the difference between the recall and precision values obtained by performing or not a filtering process, we gather the results of figure 37 and figure 44 into one figure (figure 45). We also add two curves representing the F1-scores obtained before and after the post-treatment step. The two F1 curves illustrate more the score of accuracy of the proposed approach as it take into consideration both the recall and precision scores.

The dark green line represents the recall rates and the light green line represents the precision rates without performing a filtering step. The dark blue line represents the recall rates and the red line represents the precision rates in the case of applying the connected components filtering step. We remark that the recall rates are almost the same between N=5% and N=30%. Then, the recall values obtained when we apply a filtering step are lower than when we do not process the filtering step. This is due to the fact that the connected components technique permits eliminating some text areas totally detected before the filtering step. Nonetheless, we notice that the two generated precision values are almost the same. When we apply the connected components technique, we eliminate a lot of false errors so the precision rates decrease accordingly.

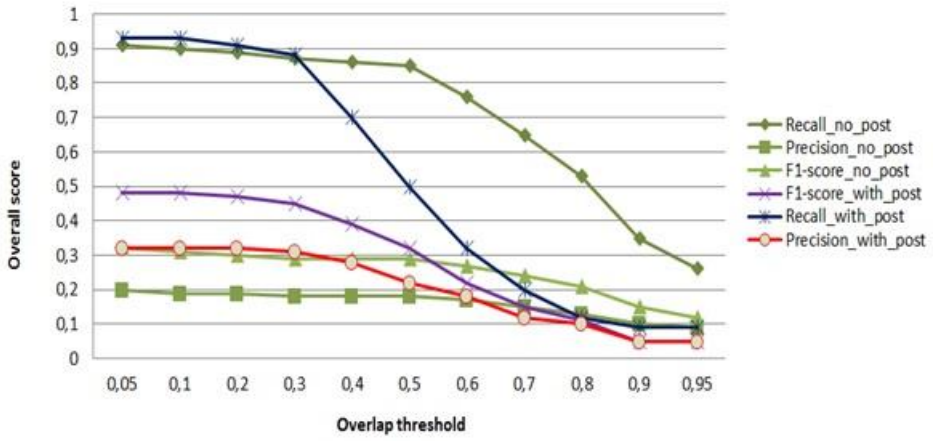


**Figure 45: The overall evaluation of text extraction. The threshold is varying from 5% to 95%.**

Finally, we conclude that our approach generates the best recall and precision rates of R=93% and P=32% respectively when N varies from 5% to 30%. We fix the overlap threshold value at 30% for the comparative study detailed in the next sub-section (sub-section 4.3.2.3).

#### 4.3.2.3. A comparative study with the state of the art

In this section, we are going to compare our results with results obtained by some methods from the literature. We select the Arai's method (K. Arai, H. Tolle 2011), the sequential information extraction method proposed in (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011), an independent information extraction method introduced in (C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, and J. M. Ogier 2013), and finally the knowledge driven analysis method presented in (C. Rigaud, C. Guérin, D. Karatzas, J. C. Burie, and J. M. Ogier 2015).

The evaluation of text localization for these methods is performed on 4691 text lines of the eB-Dtheque database (C. Guerin, C. Rigaud, A. Mercier, F. Ammar-Boudjelal, K. Bertet, A. Bouju, J. C. Burie, G. Louis, J. M.Ogier, A. Revel 2013) at object bounding box level (sub-section 4.3.2).

##### 4.3.2.3.1. Arai's method

Arai's method presented in (K. Arai, H. Tolle 2011) is considered as a sequential approach requiring the extraction of panels and balloons before localizing the text in comics. This approach is applied on grey-scale Japanese Manga. This approach was proposed to only handle the extraction of vertical text. However, it has been adapted to also handle the extraction of horizontal text by (C. Rigaud, C. Guérin, D. Karatzas, J. C. Burie, and J. M. Ogier 2015). The authors have switched the width and height and the kernel related parameters to the pre-processing step that consists in an adaptive bi-level segmentation and mathematical morphology to group text into blocks.

This method yields to a recall rate of 2.81%, a precision rate of 1.63%, and a F1-score rate of 2.06%.

#### 4.3.2.3.2. A sequential information extraction method for comics

This approach was presented in (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011) and it is considered as a bottom-up approach for text extraction in comics. This approach starts from text areas and panels in order to initiate further processing. Each processed comic image may be classified into 3 classes: “Text”, “Noise”, or “Frame”. Besides, this approach relies on a binarization process before detecting text areas. It is based on the Connected Component analysis step for text detection. It works only when the processed page contains text where the background color is similar to the paper background. However, it is able to extract text inside and outside the speech balloons.

This method yields to a recall rate of 54.91%, a precision rate of 57.15%, and a F1-score rate of 56.01%.

#### 4.3.2.3.3. An independent information extraction method for comics

This approach was presented in (C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, and J. M. Ogier 2013) and it is the continuity of the work presented in (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011). This approach consists in different steps: a bi-level segmentation step, a text/graphic separation step, a text line generation step, and a text line recognition step. For the text extraction step, the method groups text components (e.g. attached letters, isolated letters, words) into text lines. This approach overcomes the limitation presented in the sequential information extraction method presented in (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011) and the information extraction methods can be used simultaneously.

This method is evaluated in two phases: (1) in the case of the text line generation where it yields to a recall rate of 67.21%, a precision rate of 41.54%, and a F1-score rate of 51.35% and (2) in the case of text recognition step (with using an OCR system) where it yields to a recall rate of 64.14%, a precision rate of 70.28%, and a F1-score rate of 67.07%.

#### 4.3.2.3.4. A Knowledge driven approach for comics

The text extraction technique in this knowledge driven approach proposed in (C. Rigaud, C. Guérin, D. Karatzas, J. C. Burie, and J. M. Ogier 2015) is similar to the one presented in (C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, and J. M. Ogier 2013). The contribution of this approach to text extraction in comics is the ability of validating or rejecting text candidates using an expert system that include two models: (1) an Image model permits formalizing the raw data from algorithms and (2) a Comic model permits modeling the knowledge of the comics domain. A candidate text zone is validated by the system when it is contained in a panel, and it is rejected otherwise. Besides, a Tesseract Optical Character Recognition system (OCR) that has already been trained with different font and language (R. Smith 2007) is used in order to filter out regions where no alphanumerical symbols were recognized.

This method yields to a recall rate of 39.99%, a precision rate of 64.88%, and a F1-score rate of 49.48%.

#### 4.3.2.3.5. Comparison and analysis

The following table resumes all the quantitative results in terms of recall, precision, and F1 score rates of the previous mentioned approaches for text localization in comics (sub-section 4.3.2.3.1., 4.3.2.3.2, 4.3.2.3.3, 4.3.2.3.4) and of our proposed approach.

Table 4: Text localization results in comics

| Approach | R (%) | P (%) | F (%) |
|---|---|---|---|
| **Arai's (K. Arai, H. Tolle 2011)** | 2.81 | 1.63 | 2.07 |
| **The sequential (C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier 2011)** | 54.91 | 57.15 | 56.01 |
| **The independent (C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, and J. M. Ogier 2013) without OCR system** | 67.21 | 41.54 | 51.35 |
| **The independent (C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, and J. M. Ogier 2013) with OCR system** | 64.14 | **70.28** | **67.07** |
| **The knowledge-driven (C. Rigaud, C. Guérin, D. Karatzas, J. C. Burie, and J. M. Ogier 2015) with OCR system** | 39.99 | 64.88 | 49.48 |
| **Proposed** | **93** | 32 | 47.62 |

Our proposed method for text detection and localization in comics is a global method that permits detecting text areas by applying asymmetric Haar patterns. It does not rely on any segmentation technique *a priori*. The application of Haar pattern leads to detect only horizontal or vertical areas. That permits detecting only text areas that are more or less horizontal or vertical. However, curved texts are not localized totally in our context. In this case, rotated Haar filters are recommended. This affects the recall rate of our approach that is of 93%. Besides, as shown in Table4, the precision rate of our approach is low since a connected component labelling algorithm is applied in order to extract candidate text regions. As the text has most of the time a small size in the comics compared to the size of panels, speech balloons or comic characters, the CC technique groups several false negatives into components which makes them considered as candidate text zones. So, applying an OCR system to recognize text from other non-text elements is recommended to increase the precision rate. This is considered as one of our perspectives.

Furthermore, Arai's method is a sequential top-down approach. Text localization is performed after extraction of all panels and speech balloons within the comics. So, the quality of these extraction phases affects the accuracy of the text localization results.

Moreover, the sequential bottom-up method does not rely on any panel or speech balloons extraction step. However, the latter step is performed at the same time as the panel extraction step. In contrast with the sequential method, the text localization step in the independent approach is done autonomously. The use of an OCR system decreases slightly the recall rate but it increases significantly the precision rate from 41.54% to 70.28%. The drop of the recall rate generated by the Knowledge-driven method is explained by the fact that this method is based on two models that reject any detected text which is not contained in a speech balloon. Nonetheless, this method generates a good precision rate as it is based on an expert validation system. The OCR also plays a major role in increasing the precision rate.

# 5. CONCLUSION

In this chapter, we have presented an automatic text detection and localization approach for comics. This approach is based on applying Haar-like filters to detect text areas and on a connected component labeling algorithm to extract text areas. The originality of this work is that it does not rely on assumptions about the localization and the written color and style of the text in the processed comics. Indeed, this approach is done globally on the processed comic image without applying frame or speech balloons detection techniques. Experiments show that our work overcomes numerous assumptions cited in the literature that are presented previously. Besides, the best recall rate in text localization process is obtained by our work. This is because our approach is able to detect the majority of text regions within the processed comic pages. The false positives influence the precision value. Thus, the main objective of our future work will be the improvement of the precision without dropping the recall value.

*"Every great advance in science has issued from a new audacity of imagination."* **John Dewey**

# V. A coarse-to-fine analytical word spotting approach

## Contents

# 1. INTRODUCTION

In document analysis and information retrieval, the automatic study of handwritten documents is considered as a very tough task to be done. This is due to the high variability of information representation. Indeed, the access to the content of this type of documents is linked to text transcription or text recognition. Actually, the performance of the Optical Character Recognition (OCR) engines is still too low for handwriting recognition. OCR techniques are not well suited for handwriting and degraded ancient document collections in an open vocabulary context. Nevertheless, word spotting is considered as an alternative to OCR techniques for various applications such as indexing and information retrieval in digitized documents.

Word spotting tends to find multiple occurrences of a query, which is usually, expressed either as a word in ASCII form or as an image example, in document images. While analyzing occurrences of queries in historical or modern document collections, word spotting makes tasks such as indexing and information retrieval easier than without any automatic help. Besides, it can be efficient when documents are relatively complex, degraded and arduous to be retro converted to ASCII form.

Thus, numerous researches have been focusing their studies and efforts these last years on word spotting task. The architecture of every word spotting framework differ accordingly to the type and characteristics of the manipulated documents. In the literature, Word Spotting approaches have been applied to various scripts such as Latin, Arabic, Greek, etc. These scripts are not the same and they differ from each other by the number of characters, writing direction, cursiveness and similarity among letters. Moreover, documents can be either handwritten or typewritten. (for more details, you can check out the chapter of the state of the art (chapter II).

As our work is designed for manuscript documents, then the proposed approach has to be able to manipulate different types of documents especially handwritten collections that present a wide variability at different levels. These documents may have fragmented characters, variability in writing style, overlap of components like components belonging to several lines of the text because of the presence of ascenders and descenders. In our context, these documents are Latin scripts which are based on Latin alphabet. One of the most challenging constraints in a manuscript document is the variety of writing styles. For the sake of completeness, we will highlight the variability of writing styles within manuscript documents in the next sub-section. Then, we are going to describe our approach in three sub-sections: the first sub-section (3.1) details a technique that allows modeling string query word for a QBS representation in the WS process. The second sub-section (3.2) describes the global filtering module that is considered as an important module in our approach. The global filtering modules result in many candidate words pretty much similar to the searched query. To refine these results, we introduce in the third sub-section (3.3) a refining filtering module that permits obtaining more accurate spotting results. Finally, we conclude by evaluating our WS approach in section 4 and by concluding our work in section 5.

# 2. WRITING STYLES IN MANUSCRIPT DOCUMENTS

## 2.1. Letters

The proposed word spotting approach is basically based on Latin scripts that use Latin alphabet. So, it seems very important to take into consideration the shape of each letter. This latter may vary within the same text or in a word wherever its positions in the word.

## 2.2. Variability of the writing styles

As we mentioned above (Chapter I and chapter II), one of the major constraint of word spotting in manuscript documents is the variation of the writing style. Here, for the sake of completeness, we are briefly going to describe the variation in writing styles context.

The variation in writing styles is defined as the manner how a writer chooses to express him or herself through writing.

The different writing styles represent different properties such as:

- Unique for each writer
- Formal and informal writing styles: each style has its own purposes as mentioned above
- The writing of letters and digits varies considerably from one writer to another.
- The writing styles vary from one century to another (i.e. differ during centuries)
- It may be regular or irregular, neat or tormented, clear or not
- Depends on the handedness of the writer (i.e. left-handedness, right-handedness, mixed-handedness, and ambidexterity)

More other properties are mentioned in paleography science article such as (A. de Bouard 1911) (J. J. Marcos 2011)

Furthermore, in order to show more precisely how the writing styles differ based on the mental and psychic states of each human being, we illustrate in the next two figures (figure 46 and figure 47) some examples. These examples contain signature and writing notes that have been written by left and right handedness students before and after doing an exam.


**Figure 46: Examples of manuscripts written by different writers before and after an exam. The left column contains notes written before an exam. Each notes is written by the same writer after the exam (right column)**

Besides, in the next figure (figure 47), each writer was asked to write the same template (figure 46 before exam) after 3 months. We remark that the obtained templates differ from the first ones even if there is no stress factor influencing the writers. Indeed, as evidence, no one is able to control his own writing style amongst too many life variables.


**Figure 47: Examples of manuscripts written by different writers (left column) and after 3 months later in the same conditions from that exam (right column).**

We conclude that each human being has its own writing style. This latter may differ according to the different constraints that have been mentioned in the beginning of this sub-section. These illustrations show the major constraint of all manuscripts that is: The variability of the writing styles. This is considered as a tough challenge that has to be overcome during spotting queries in manuscript documents.

# 3. THE PROPOSED WORD SPOTTING APPROACH

The proposed word spotting approach can be justified due to the fact that it is essentially based on human perception characteristics, field that occupies an important position when visualization and image understanding is considered. We were inspired by two characteristics of human vision (Treisman 1985) (R. W. Proctor, K. P. L. Vu 2003):

- Preattentive processing that led us to develop a coarse-to-fine structure.
- The principle of the human perception which assumes that first the different individual features affect perception then these features are gathered together to obtain object notions.

These characteristics help us to introduce a multi-scale approach that is propagating from page level at coarse scale to fine scales and to multiply the number of viewpoints allowing focusing on relevant parts in processed documents.

Besides, another contribution of our project is the ability to manipulate different types of documents especially handwritten collections that present a wide variability at different levels. These documents may have fragmented characters, variability in writing style, overlap of components like components belonging to several lines of the text due to the presence of ascenders and descenders. Taking into consideration these constraints, we introduce an original approach which is based on:

- No layout segmentation of the processed documents: there is no segmentation and the query is searched along the document.
- No binarization step of the processed documents to avoid losing data preventing from discriminating roughly similar words.

The fundamental principle of the proposed model is based on a coarse-to-fine approach. The whole document is first scanned in order to have a global view that is nevertheless biased toward the element (e.g., query) that is looked for. This step enables to focus the attention on some parts of the document where a more specific observation will be achieved. Then, at a lower scale, we attempt to find the exact query in processed documents with another set of properties. This allows reducing the number of false positives (e.g., observations that are very similar in form with the query) and the results will be much better at this observation scale. (For more details, please take a look at subsections 3.1 and 3.2).

Our coarse-to-fine approach is an analytical one that is applied on the entire processed document images without any use of segmentation techniques. Because it is analytical, the computation time is greater than when the hypothesis of a prior localization of words is made. Taking into consideration that information, we propose an analytical word spotting approach that is based on:

- Word coding step for string queries representation (sub-section 3.1).
- Two filtering modules that are performed sequentially at two different observation scales (sub-sections 3.1 and 3.2).

The two filtering modules are as follows:

- A global filtering, to reduce the search space into zones of interests (ZOIs) containing candidate words (CWs) (sub-section3.2).

- A refining module, to retain only candidate words which are the most similar to the query (sub-section 3.3).

The analytical approaches permit segmenting the entire documents or words into smaller units that will be recognized when they are isolated or grouped. Besides, analytical approaches may be performed on the entire document without processing any segmenting step. Thus, our approach will be first applied on the entire document without segmenting it into words or text lines or smaller units. This represents the first filtering module that permits generating zones of interests that are considered as potential candidates for the input string query. Then, the observation scale is changed, and the method will be applied on the neighborhood of these zones of interest. Thus, the observation scale is changing from top to bottom. In fact, it changes from document scale to word scale.

The flowchart of our analytical coarse to fine approach is illustrated in figure 48.

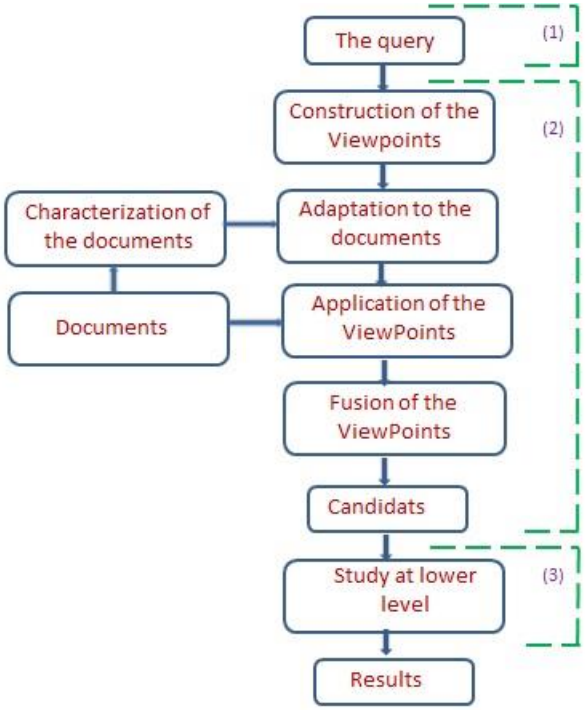


**Figure 48: The flowchart of the proposed word spotting approach. (1) The word coding step for string queries representation. (2) The global filtering module. (3) The refining filtering module.**

Furthermore, we have reviewed previously that most word spotting approaches are based on the query representation. Either they are based on Query-by-string (QBS) representation or on Query-by-example (QBE) representation. In the conclusion section of the state of the art of word spotting approaches, we have chosen to base our work on the QBS representation due to the fact that this type of representation enables to look for string queries that may not be existed in the concerned documents. Thus, we are going now to propose a novel technique that permits representing the query typed by a user in an ASCII form.

In the following sub-sections, we are going to describe the modeling word technique for string query representations, then, we are going to detail the global and refining modules.

## 3.1. Modeling word technique for QBS representation

The QBS methods (H. Cao and V. Govindaraju. 2007) use character sequences as input. It typically requires a large amount of training materials since characters are learned *a priori*. The model for a query is built at runtime from the models of its constitutive characters.

Furthermore, there are several techniques to generate the query in the QBS based approaches. Some of these techniques produce word images as a query for the system. In Latex, different types of fonts may be used to generate synthetic word image from the ASCII query given by the user (see sub-section 3.1.2.2.1. of section 3 in chapter II). However, we will not base our work on these techniques, but we will introduce a novel technique that permits modeling every ASCII text typed by the user in the form of successive rectangular shapes.

## 3.2. Rectangular-shape coding technique for string queries representation

First of all, let's consider this question: what's the easiest and simplest way to represent a list of successive characters and punctuations?

Indeed, if we focus our attention on various visualized words, then our visual system may discriminate those words one from another by their corresponding shapes. If they differ enough, then only a rough description is sufficient. Precisely, each connected letters, consisting of all written symbols, can be characterized or embedded by rectangular connected shapes. The next figure resumes this observation (figure 49).


**Figure 49: Modeling strings by rectangular shapes. (a) The user typed string (b) Black rectangles characterize the written letters, and the white one characterizes the blank space.**

Hence, every input string may be coded by a sequence of connected rectangular shapes. This coding technique is asymmetric; that means that if we know the string then we may define exactly its generated code, however, the reverse is not correct. We should mention that a recognition process after this coding step is not crucial because we will not need this recognition process in our word spotting framework.

Yet, instead of converting the typed string into a synthetic image, we propose to convert it into different rectangular shapes.

Thus, the idea is to construct a look up table where the input consists of different classes representing written symbols, and the output will be the associated rectangular shapes. Once this look up table is created, we can code every typed string by connecting rectangular shapes associated with each character that compose this string. To ensure the consistency of the look up table, we are not going to discriminate between lower and upper case letters. Besides, we classify the written symbols by their characteristics (letters with ascenders, letters with descenders, punctuations, etc.). The look up table has hundreds of inputs classified into 5 cases and with each case different shapes are associated.

The look up table contains five main classes as input:

- Class 1: Upper case letters
- Class 2: Lower case letters
- Class 3: Numbers
- Class 4: Punctuations
- Class 5: Blank space

Furthermore, we classify lower case letters into 4 sub-classes as following:

- Case 1: letters with ascenders as: ‘b’, ‘d’, ‘h’, ‘k’, ‘l’, ‘t’, ’f’.
- Case 2: letters with descenders as : ‘g’, ‘j’, ‘p’, ‘q’, ‘y, ‘z’.
- Case 3: letters with both ascender and descender as: ‘f’.
- Case 4: letters as: ‘a, ‘c’, ‘e’, ‘i’, ‘m’, ‘n’, ‘o’, ‘r’, ‘s’, ‘u’, ‘v’, ‘w’, ‘x’.

To clarify more the association of the letter ‘f’ in apart case, it is because in various handwritten contexts, this letter is written with both ascender and descender, however, it may be also be written with only an ascender. Besides, lower case letters with diacritics (i.e. accent, circumflex, marks, etc.) are considered as lower case letters in our work.

In order to visualize the look up table, various rectangular shapes, which differ in terms of size and color, are associated with each class and sub-class within the look up table. The size of each corresponding shape is strongly depending on the size of the written characters in the tested document in order to well adapt the generated coding results with the characteristic of the processed document.

After building a priori the look up table, each input query sequence is rectangular-shape coded automatically. The next two figures (figure 50 and figure 51) show two different examples of this coding process.

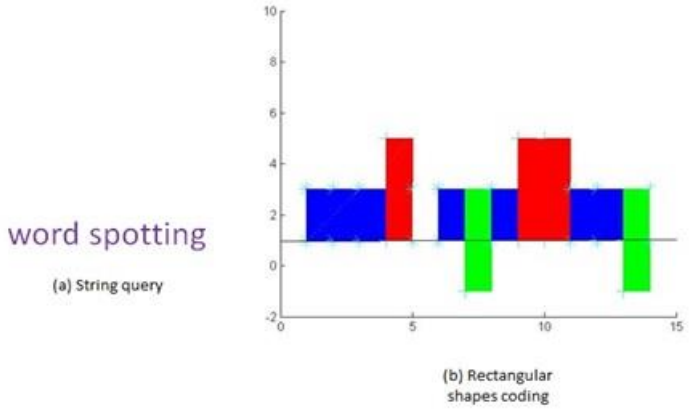


**Figure 50: Coding a query by rectangular shapes. (a) The typed query that is composed by two words (b) blue rectangles represent letters without ascender or descender, the red ones represent letters with ascender, and the green ones represent letters with descenders.**

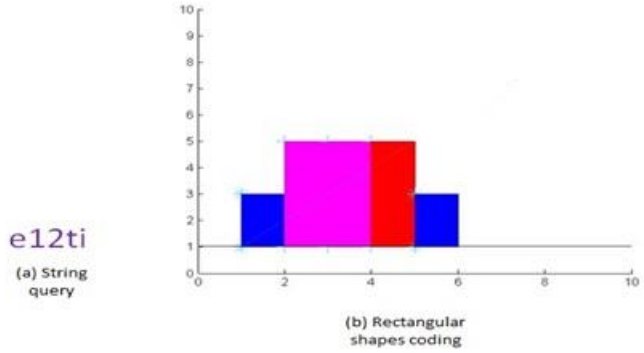


**Figure 51: Coding a query by rectangular shapes. (a) The typed query that is composed by characters and numbers (b) blue rectangles represent letters without ascender or descender, the red one represents letter with ascender, and the magenta ones represent numbers.**

Furthermore, as we mentioned earlier, the size of these rectangular shapes are depending on the size of the written characters in each manipulated document in order to adapt the size of typed query with the average size of the writing, so we shall find a relationship in term of size between the different elements. The idea is as following: An Index Numeric Table (INT) is generated automatically in the coding process. Each column of this table contains an index where each index corresponds to a proportional number between the size of the desired obtained rectangular shape and the default size of a lowercase letter without any ascender or descender. To simplify this, we give a simple example. Most of the time, the size of a written character with an ascender (represented as SA) is two times higher than the size of a simple character (represented as SC), which no ascender or descender. So, we may model that by the following relation:

$$SA = 2 * SC \qquad (23)$$

Thus, the distribution of the different proportional indexes of each class in the look up tables is as following:

- Class 1: Upper case letters: 2
- Class 2: Lower case letters
- Class 3: Numbers: 2
- Class 4: Punctuations: 1
- Class 5: Blank space: 0

And for the sub-classes of the Class 2, it goes as following:

- Case 1: letters with ascenders: 2
- Case 2: letters with descenders : -2
- Case 3: letters with both ascender and descender: 4
- Case 4: letters without ascender or descender: 1

Figure 52 and figure 53 show two different examples of this coding process. The difference between these two figures is that we show an example (figure 53) of a string query composed by a letter with both ascender and descender and a letter with descender.

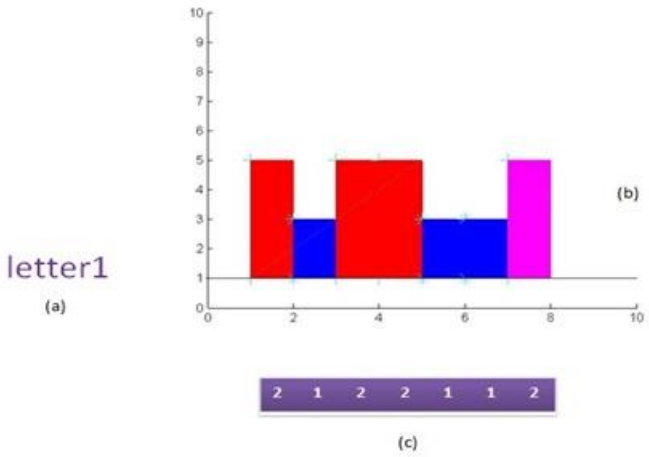


**Figure 52: The coding process (a) The typed query that is composed by characters and a number (b) blue rectangles represent letters without ascenders or descenders, the red one represents letter with ascender, and the magenta one represents numbers. (c) The generated index table. The number 2 represents letters with ascenders or numbers; the number represents lowercase letters without ascenders and descenders.**

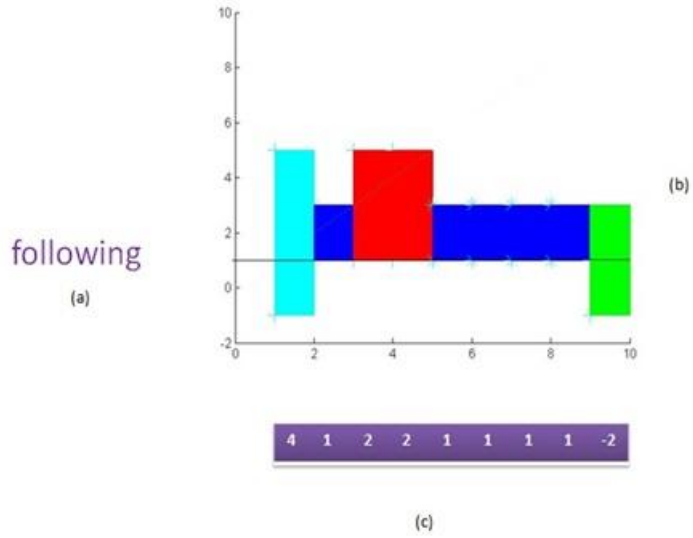


**Figure 53: the coding process (a) The typed query that is composed by characters (b) blue rectangles represent letters without ascenders or descenders, the red one represents letter with ascender, the green one represents letter with descender, and the cyan ones represent letters with ascender and descender. (c) The generated index table. The number 4 represents letters with ascender and descender, the number 1 represents letters without descenders or ascenders, the number 2 represents letters with ascenders, and the number -2 represents letters with descenders.**

One could say that it is not necessary to discriminate between the indexes of uppercase letter and letter with ascenders because an uppercase letter has approximately the same height as a letter with ascender and then they are considered as the same, identically for the index indicating numbers. As most of the time, writers tend to write numbers with the same height as an uppercase letter. Indeed, distinguishing them makes the model technique of the string query more accurate as it takes into consideration the different elements of Latin alphabet.

Moreover, the generated indexes point out the position of each component in the input string query. Thus, we obtain an abstract representation of the query.

Hence, the look up table will play an essential role that addresses accurately the Query-by-String problem.

The word coding process for string queries representation is the first phase of the proposed analytical approach. Now, we are going to introduce first the global filtering phase and then the refining phase of our proposed approach.

## 3.3. Global filtering module

The global filtering step represents the essential and original step of the analytical word spotting proposed approach. The global filtering aims to reduce the number of zones of interest corresponding more or less faithfully to the given query. In this phase, we propose to apply a certain number of viewpoints (VPs) that are treated independently and whose findings are gathered together with an accumulation vote process to make final results. Indeed, the different steps of this filtering module are illustrated in Figure 54.

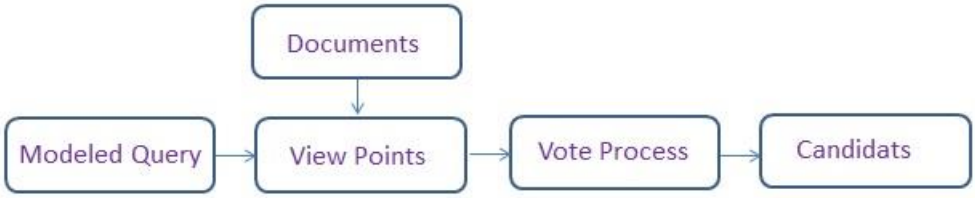


**Figure 54: The flowchart of the different phases that define the global filtering module.**

Furthermore, these viewpoints are built according to the way human perception reacts with respect to the shape of a word, focusing for instance on the global shape of a word or to the more salient parts like ascenders and descenders. Indeed, a viewpoint corresponds to a global transformation applied to the processed document allowing highlighting different shapes. However, these filters must be depending on the query and document properties. Hence, the choice of the type of filters is essential and must achieve a tradeoff between efficiency and fast computation time.

Besides, these viewpoints are highly dependent on the query and documents properties for each manipulated document. These proposed filters should be constructed taking into account two types of information:

- The information concerning the query we are looking for in documents.
- The information concerning the entire processed document images that is the style of the writing.

Subsequently, our choice was the use of Haar-like features that are easily computed from the integral image (P. Viola, M. Jones 2001b) of each processed document image. These filters are more simple to be used than Haar wavelet coefficients used in different document applications such as document text extraction (S. Audithan 2009) or script identification (P. S. Hiremath, S. Shivashankar 2008). With the use of wavelet, computation is done at different scale levels and the best ones are to be selected according to criteria that are defined according to the application. In our case, the selection of the right filters and the right scale has to be achieved according to the word and to the characteristics of processed document images. Indeed, a word is characterized by intrinsic and exogenous characteristics such as the number of letters, the position and presence of ascenders and descenders, the size (width and height) of each character, etc. Thus, we have defined generalized patterns to fit the global shape of words written in Latin alphabet.

Figure 55 shows a set of generalized Haar-like features that are used in our work.

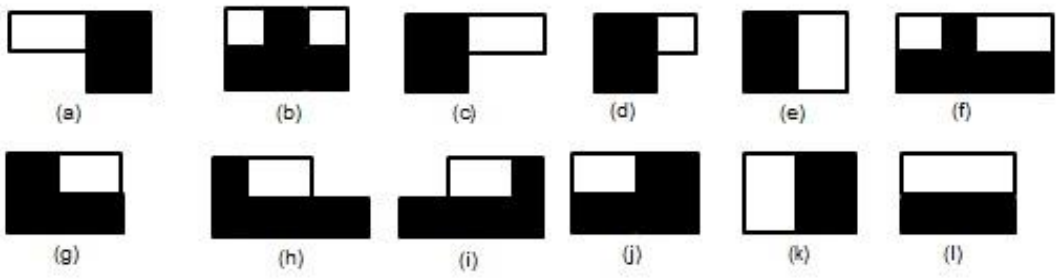


**Figure 55: Few generalized Haar-like filters applied in our work. Each pattern may characterize a portion of the string query word. For instance, in (g), the black shape represents a letter with ascender followed by a letter without**

**ascender or descender and the white shape represents the background. In (b) the black shape represents succession of lowercase letters without ascenders or descenders with a letter with ascender in the middle.**

The generalized Haar-like features are asymmetric. The black and white areas have not always the same heights and widths. This depends on the nature of the string query. The black or white shapes may be composed by two rectangles or by a set of successive rectangles. Besides, each pattern is constituted by a black shape and a white shape where:

- The black area represents the writing
- The white area represents the background of the document image

Each pattern will fit a portion of the string query looked for. It defines precisely the shape of this portion. For instance, in figure 55 (h), the black shape represents a letter with ascender followed by sequence of successive lowercase letters without ascenders or descenders and the white shape represents the background or the non-ink area. Besides, in figure 55 (i), the black shape represents a lowercase letter without ascenders or descenders and lastly a letter with ascender. Hence, every pattern has its own property that helps us in detecting regions constituted by, for instance, an uppercase letter at the beginning followed by a certain number of lower case letters, a lowercase letter without ascender and descender followed by a certain number of letters with ascenders or descenders or both of them, etc.

However, the Haar-like filters have to characterize the string query written by the user. Yet, we are going to detail to process of the construction of generalized Haar-like filters in an automatic way.

### 3.3.1. The construction of the generalized Haar-like features

As it is shown in the flowchart of the global filtering module (figure 54), the construction of Viewpoints has to take into account the characteristics of the query. Yet, these two phases (sub-section 3.2 and sub-section 3.3.1) are related. Thus, we are going to detail the relation between these two phases.

Indeed, the look up table generated by the Rectangular-shape coding technique table indicates the nature and the relative position of each element in the string query. Consequently, we may define a connection between the query and the Haar-like filters. Besides, the INT suggests different clues for constructing generalized Haar-like filters. These clues determine:

- The successive indexes of the INT indicate how to fit the string query by the applied GHF. This permits focusing on only the candidate regions that look similar to the query in the manipulated document. Figure 56 illustrates an example of this clue where couples of successive characters are considered.
- Each index indicates the height of every GHF. The determination of the height value is considered as an absolute constraint for the process of word spotting in manuscript documents. The height cannot be *a priori* determined because we manipulate handwritten documents. Hence, it should be automatically determined taking into consideration the characteristics of the writing of each manuscript document.
- The number of GHF that would be applied for searching the query in the document. Experiments show that the sufficient number of GHFs that will be applied for each query of size greater than 3 is 4. Otherwise, 2 GHFs are applied for queries of size smaller than 3. These GHFs will take into consideration the characteristics of characters representing the beginning, the end, and the middle of each query.

Moreover, these clues are essential to construct automatically the GHFs. We are going to write the pseudo-code that permits generalizing automatically the Haar-like filters according to the spotted query (Table 5):

**Table 5: The pseudo-code of GHFs construction**

```
Input : INT [1..n] %% Index Numeric Table
                   %% indx(INT(1)) means the element in case number
1 in the INT
Output : GHFs %% the generalized Haar-like filters

Algorithm

L<-length(INT) %% the length of the INT

GHFbegining<-
DrawRectangle(indx(INT(1)))+DrawRectangle(index(INT(0))) %%
DrawRectangle(indx(INT(1)))means that a Black Rectangle with the
size of (indx(INT(1)) will be drawn and DrawRectan-
gle(index(INT(0)))will be a default white rectangle with the same
size as the associated black one.


GHFend<-DrawRectangle(indx(INT(n)))+DrawRectangle(index(INT(n+1)))
%% DrawRectangle(indx(INT(n)))means that a Black Rectangle with the
size of (indx(INT(n)) will be drawn and DrawRectan-
gle(index(INT(n+1)))will be a default white rectangle with the same
size as the associated black one.

If(L % 2 = 0) %% in this case the length of the INT is an EVEN num-
ber

GHFmiddleRight<-
DrawRectangle(indx(INT(L/2)))+DrawRectangle(index(INT(L/2+1))) %%
DrawRectangle(indx(INT(L/2)))means that a Black Rectangle with the
size of (indx(INT(L/2)) will be drawn and DrawRectan-
gle(index(INT(L/2+1)))will be drawn taking into consideration the
index(INT(L/2+1))

GHFmiddleLeft<-
DrawRectangle(indx(INT(L/2)))+DrawRectangle(index(INT(L/2-1))) %%
DrawRectangle(indx(INT(L/2)))means that a Black Rectangle with the
size of (indx(INT(L/2)) will be drawn and DrawRectan-
gle(index(INT(L/2-1)))will be drawn taking into consideration in-
dex(INT(L/2+1))

ENDIF

If(L % 2 != 0)  %% in this case the length of the INT is an ODD
number

GHFmiddleLeft<-DrawRectangle(lower integer
of(indx(INT(L/2))))+DrawRectangle(lower integer of (index(INT(L/2-
1)))) %% DrawRectangle(lower integer of (indx(INT(L/2)))means that
a Black Rectangle with the size of (lower interger of
```

```
(indx(INT(L/2))) will be drawn and DrawRectangle(lower integer of
(index(INT(L/2-1))))will be drawn taking into consideration the
lower integer of index(INT(L/2-1))

GHFmiddleRight<-DrawRectangle(upper interger of
(indx(INT(L/2))))+DrawRectangle(upper integer of (in-
dex(INT(L/2+1))) %% DrawRectangle(upper integer of
(indx(INT(L/2)))means that a Black Rectangle with the size of the
upper integer of (indx(INT(L/2)) will be drawn and DrawRectan-
gle(upper integer of (index(INT(L/2+1))))will be drawn taking into
consideration the upper integer of index(INT(L/2+1))

ENDIF
```

Hence, for each "rectangular-shape" coded string query, we may determine the different GHFs that fit the query. Yet, the look up table is essential for the construction and generalizing the Haar-like filters (figure 56).

The number of patterns and their shapes are in relationship with the characteristic of each input string query through the look up table. However, the height value of each pattern is still a constraint that has to be overcome.

In the next sub-section, we introduce an automatic technique for estimating the average height of lowercase handwritten letters in document images.

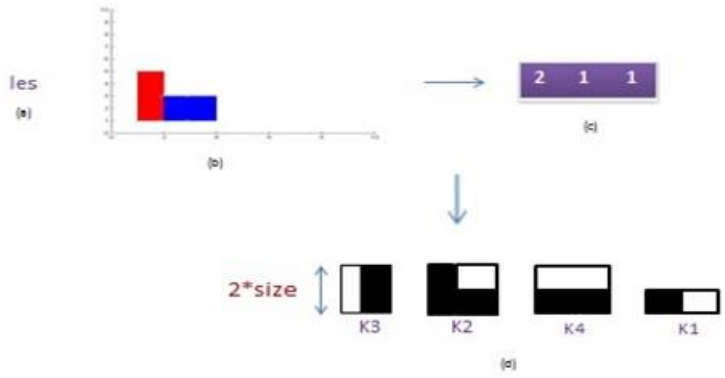


**Figure 56: An example of the construction of the GHFs. (a) the string query. (b) The generated "rectangular shape" modeling of the query. (c) The INT of the look up table. This INT contains 3 elements, thus, the number of GHFs would be 4. Those elements indicate that the searched region must begin by an uppercase letter or a lowercase letter with an ascender followed by two lowercase letters without ascenders or descenders. (d) The recommended GHFs. K1 specifies that the searched candidates must end by a lowercase letter without ascender or descender. K2 specifies that these candidate regions contain an uppercase letter or a lowercase letter with an ascender followed by a lowercase letter without ascender or descender. K4 represents a text area with two lowercase letters without ascenders and descenders. Finally, K3 affirms that there is an uppercase letter or a lowercase letter with ascender at the beginning of candidate regions.**

### 3.3.2. An automatic height estimation technique for letters in manuscripts

The construction of the different patterns is highly dependent on the size of characters which vary in accordance with the writing style of document to be processed. Thus, though the initial shape of the patterns depends on the shape of the query word itself, the size of the patterns is depending on the processed document characteristics. The reference height of the Haar filters is linked to the height of the bodyline of writing. Hence, we propose a size estimation technique. This technique estimates automatically the size of lowercase letters within each manipulated hand-written document image. Here, we must mention that the size of different characters is not unique due to variability of the writing styles. The aim is not to detect lines but to estimate the height of the predominant lines.

The core of the technique is as following:

To detect horizontal text line, experiments show, as mentioned in previous chapter, that it is possible to use only one Haar-like pattern that would be pretty much similar as the text line (figure 57 (b)). The answers to this filter will be the highest when the sizes of the filters are in accordance with text lines. Then, we apply progressively this filter by varying its size in order to generate various responses for each size. The various responses, obtained by applying the Haar-like filter globally on the input image, consist of the written regions that represent the different body lines of the text. When the size of the applied pattern increases, the height of resulted responses increases too. Indeed, the blank space between each text lines ensures the vertical discontinuity of the obtained responses. This vertical discontinuity is not ensured when the height of the applied pattern is much larger than the height of the body line of the text. After generating these responses, we compute the horizontal projection histogram for each one. The horizontal projection histogram plots the number of pixels for each text line. So, the local maximum values indicate the height of the text body lines. We remind that the objective by the determination of horizontal projections is not the segmentation of the document pages into text lines, since our approach does not rely on any layout segmentation process (see figure 57).

Moreover, for each filter application, the median of the different peaks of the horizontal projection is considered. This value y is associated with the filter size x as $y = f(x)$. Indeed, we are looking for the size where line and filter have the same size. Then, the line size estimation is obtained as $f(x) = x$. This permits estimating a suitable height of the text body lines within the original document page. Otherwise, we estimate the height of characters whatever their characteristics (uppercase, lowercase, ascenders, descenders, etc.)

This process is illustrated in the figure 57.

This technique does not reproduce the exact size of handwritten letters in the document. This latter may vary in the same document page due to the non-stability of each writing style. Hence, the obtained height represents an approximate estimation of the height value of handwritten letters.

Furthermore, in such cases, we may obtain more than one intersection between the bisector and the curve line made by the obtained median values and the corresponding size, as it is shown in figure 58. Then, we may choose one value between the resulted values. This is due to the non-stability of the writing style of the same writer that is highly depending on his/ her physical and psychical state. The hazardous choice between these intersections does not disturb the stability of final obtained responses as it is concluded by different experiments.

Finally, after the two phases of constructing the different generalized Haar-like patterns and the automatic estimation of the height of each pattern, we are going to detail the remaining phases of the global filtering module in the next sub-sections.

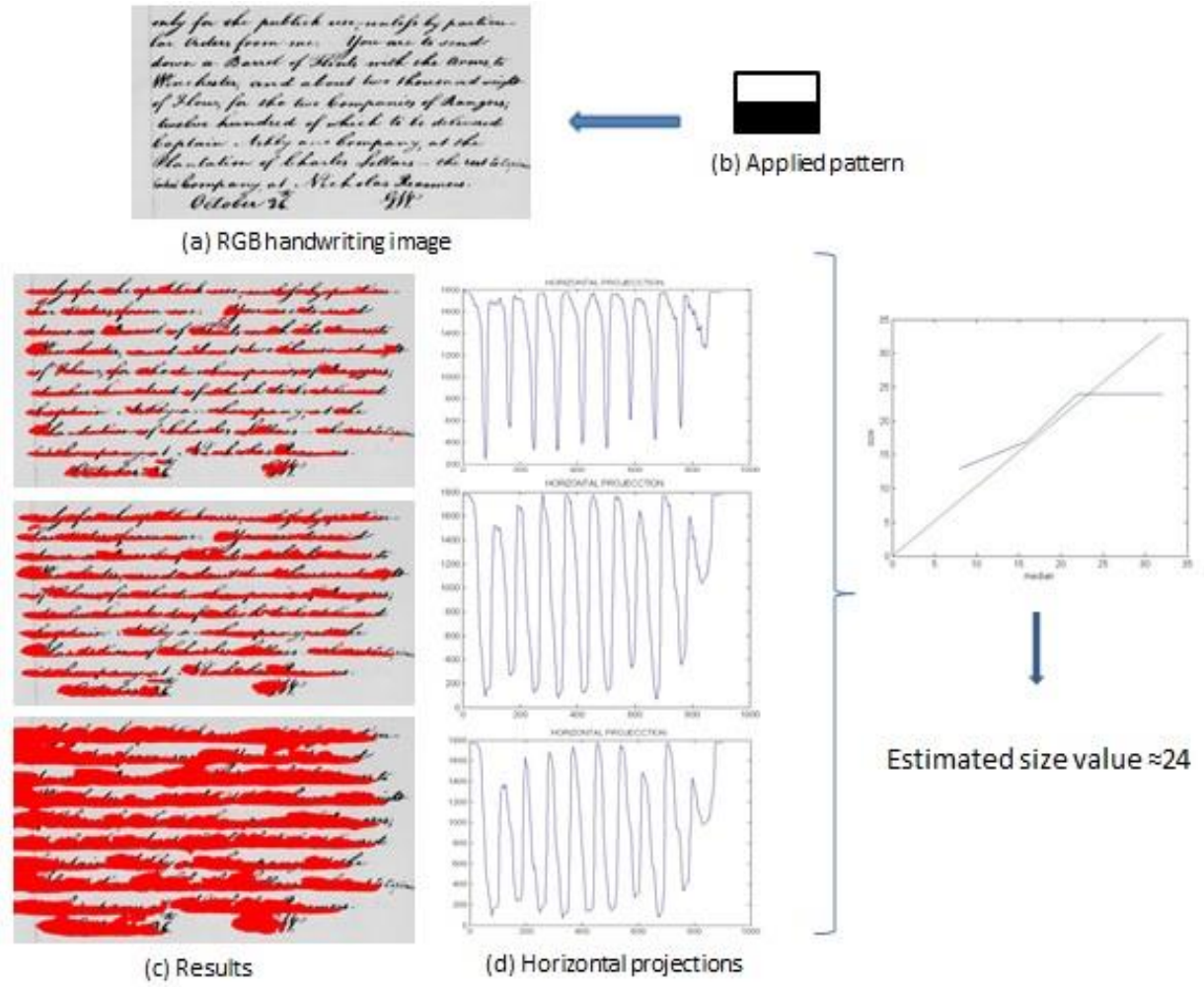


**Figure 57: The automatic height estimation technique. The applied pattern will detect the body line of each line in the manipulated RGB handwriting image. By varying its size, different responses are obtained. These responses are highlighted in red. For each result, the horizontal projection profile is generated. The estimated height is determined as the intersection between the bisector and the curve line representing the median values of peaks and their corresponding size.**

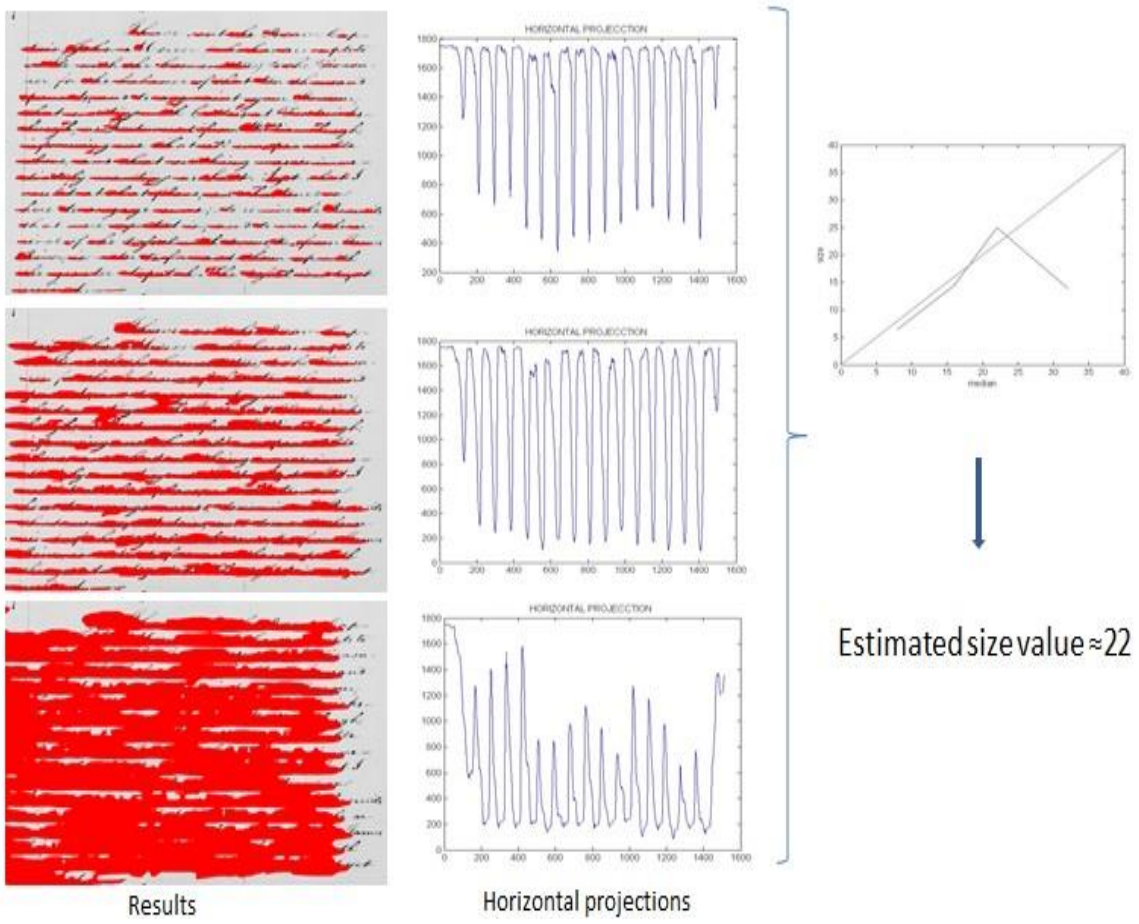


**Figure 58: An example of automatic height estimation technique that generates two intersections between the bi-sector and the curve line made by the obtained median values and the corresponding size. We obtain two values of 22 and 16.**

### 3.3.3. A vote accumulation technique

After defining the GHF that is characterized by a kernel (K), this latter is applied as a global transformation on the document leading to a transformed image $I_k$ of the processed document image Doc. Besides, the transform is a convolution operation (formula 24). This has been described earlier in Chapter IV sub-section 3.1.1.

$$I_k(p) = \iint K(p-x)Doc(x)dx \qquad (24)$$

$I_k$ highlights the spatial areas where the shape associated with the kernel is present.

This filtering process helps us in searching various ZOIs in the tested document image Doc, ZOIs that have some similarity with the query and could be linked to a CW. Indeed, in $I_k$, the value of each pixel can be interpreted as a confidence in the presence of a specific pattern in the neighborhood of the pixel. We will consider each obtained value as a vote that shows the existence of the shape in that spa-

tial area. The variety of the kernels gives partial information of the real existence of the query. The figure 59 illustrates the transformed image obtained by applying an asymmetric GHF on a given document image.

With an accumulation process of this information, we obtain ZOIs that are highly similar to the query. The accumulation process is a primordial step in our approach. In fact, the spatial location of each vote on each pixel is depending on the INT obtained by the "rectangular-shape" coding process of the string query. Hence, the INT will play a major role in the translation of the different votes in order to be accumulated.

The INT predicts the approximate length of the string query, otherwise the length of the searched candidate zones within the manuscript document image. It cannot indicate the exact length of ZOIs because the string query is typed by the user and the writing style within document pages may vary unpredictably. In addition, each index within the INT helps us to decide the spatial location of each applied GHF. Also, the INT permits generalizing Haar-like filters (sub-section3.3.1).

Let's assume that we have a string query of length 3 (figure 56 (a)). The automatic construction of the GFHs is explained in sub-section 3.3.1. We choose to accumulate the different responses in the extremity of candidates regions. Thus, the vote resulting from the application of K1 will not be translated because it corresponds to the end of the candidate region. The vote resulting from applying K2 will be translated by 2 characters, the vote resulting from applying K3 will the translated by 3 characters and finally the vote resulting from applying K4 will be translated by 1 character.

We illustrate the process of the accumulation vote technique in figure 60. This example is the continuity of the example highlighted in figure 56.

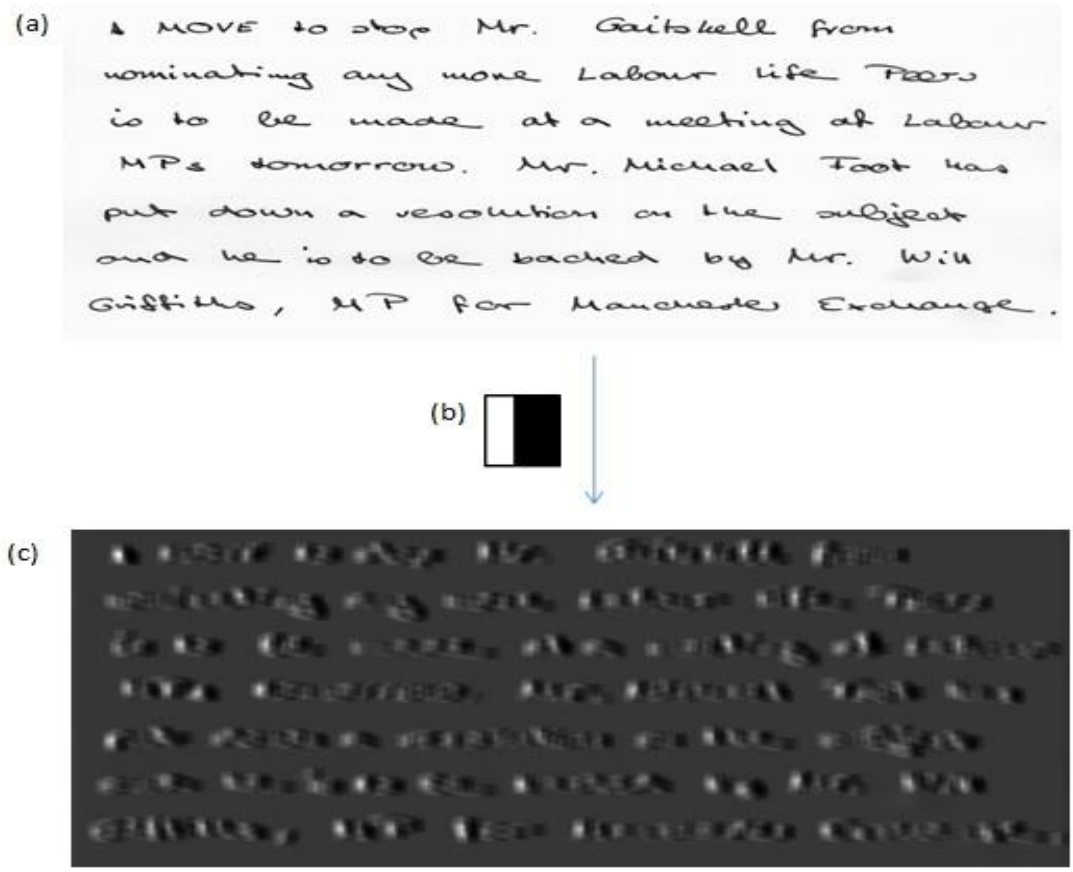


**Figure 59: The transformed grey image obtained by applying an asymmetric GHF (b) on the manuscript document image (a). (b) The applied pattern permits detecting zones of interests beginning by an uppercase letter or a lowercase letter with an ascender. (c) The transformed grey image. The potential responses are highlighted in white within this image. They vote for the presence of the shape in that spatial area.**

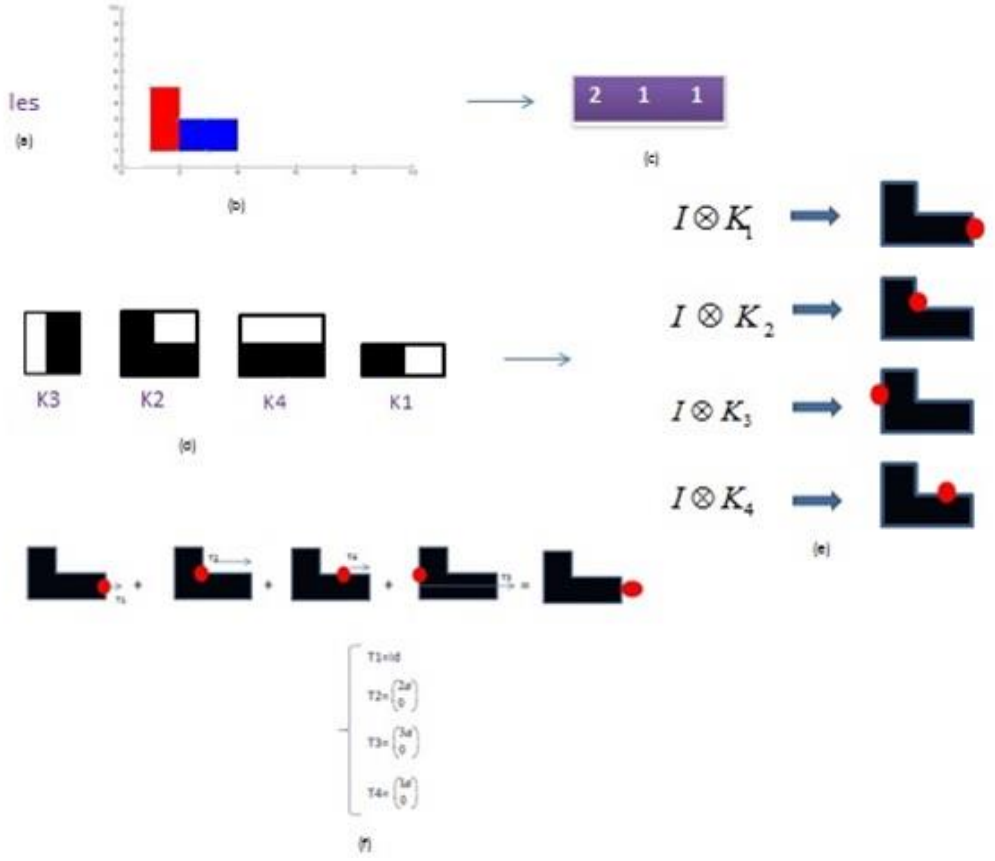


**Figure 60: An illustration of the accumulation vote technique. (a) The string query. (b) The generated "rectangular shape" modeling of the query. (c) The INT of the look up table. This INT contains 3 elements; the number of GHFs would be 4 (see figure 56). (d) The applied GHFs with their kernels (see figure 56). (e) The results obtained by the convolution operation between the GHFs and the query are highlighted in red. (f) Indicates the accumulation of the votes at the neighborhood of the end of the searched query by a translation process.**

After gathering the different votes over the potential candidate by a simple translation process, we sum all the filtered images as following:

$$I_{accum} = \sum_{k} t_k \otimes I_k \qquad (25)$$

$t_k$ represents the translation of the vote resulting by applying the Kernel K and $I_{accum}$ is a grey level image summing up the whole votes.

The final result specifying all candidate words is obtained by the binarization of the accumulated image.

### 3.3.4. The automatic threshold estimation technique for the binarization

As we are facing various constraints such as the variability of the writing style, the determination of the threshold value of the binarization process is a difficult step. Hence, we propose to obtain automatically a threshold value for binarization of each final transformation of processed document image taking into consideration its characteristics.

When we focus on each transformed grey image, we observe potential responses within it. These responses indicate the presence of ZOIs. As the objective is to binarize the accumulated transformed grey image in order to highlight the results, then, the threshold value of the binarization could be in relationship with these possible candidates. Thus, the threshold value may not be constant and fixed a priori. It varies with the variation of the document characteristics.

Indeed, the automatic threshold estimation technique for the binarization is described as follows:

In $I_{accuum}$ we are concerned by the maximum values. Then, we find all local maximums that highlight the maximum values obtained by the convolution operation between applied patterns and the manipulated manuscript document. Thereby, the threshold binarization value is taken as the average of obtained local maximum values. This means that after binarizing of the transformed grey image, we are interested only in the potential responses which values are above the threshold value.

An illustration of this process is highlighted in figure 61.

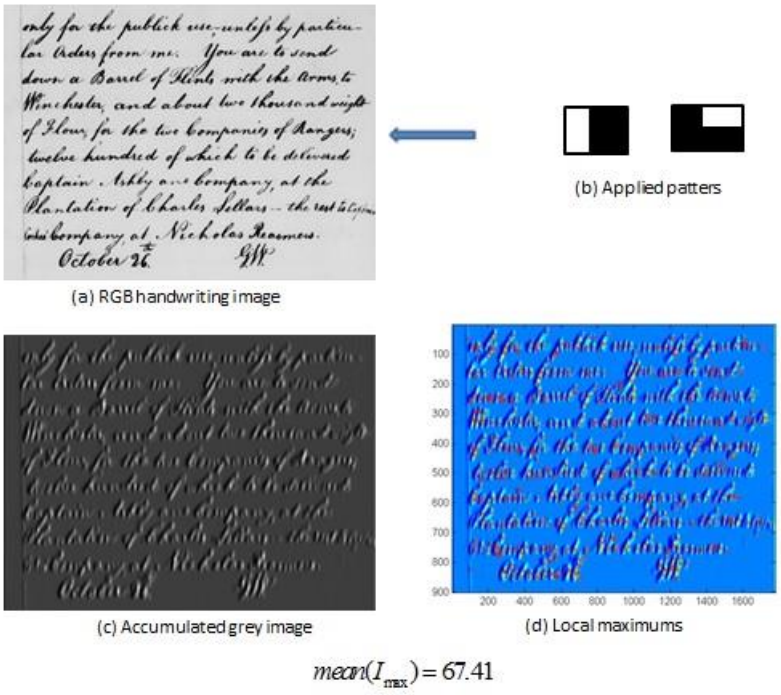

**Figure 61: Automatic estimation of the threshold value of the binarization step. Two GHFs are applied in this example. Thus results in an accumulated grey image that sums up the two transformed grey images obtained by the application of each GHF. The local maximums are computed. The threshold value of binarization is considered as the average of the local maximums.**

Finally, we obtain a reduced number of zones of interest corresponding more or less faithfully to the given query. Yet, we are in front of a high number of zones of interests, so we are going to process with vote analysis process.

In order to simplify this process, now, we are going to give an example to illustrate the accumulation process after the binarization step. Let's assume that we would like to search textual regions having some properties: they begin with an uppercase letter or a lower case letter with ascender followed by a lowercase followed by a lower case letter with ascender and they end by a 3 consecutive lower case letters (figure 62 (d)). For that, we are going to use only three GHFs in order to well highlight the accumulation results. The generated GHF are illustrated in figure 62 (b). The obtained responses are figured in colors just in order to distinguish them. We decided that the accumulation vote would be at the beginning of candidate words. In fact, the responses of the pattern are colored in green. Besides, the blue ones represent the responses of the second pattern. It is illustrated that these blue regions are translated to the left by approximately two letters. The third pattern that detects the end of candidate words generates responses drawn in red. These are also translated to the left but by approximately 6 letters. Furthermore, the pixel density regions representing the votes seem to be more crowded when the confidence in the words is more important.

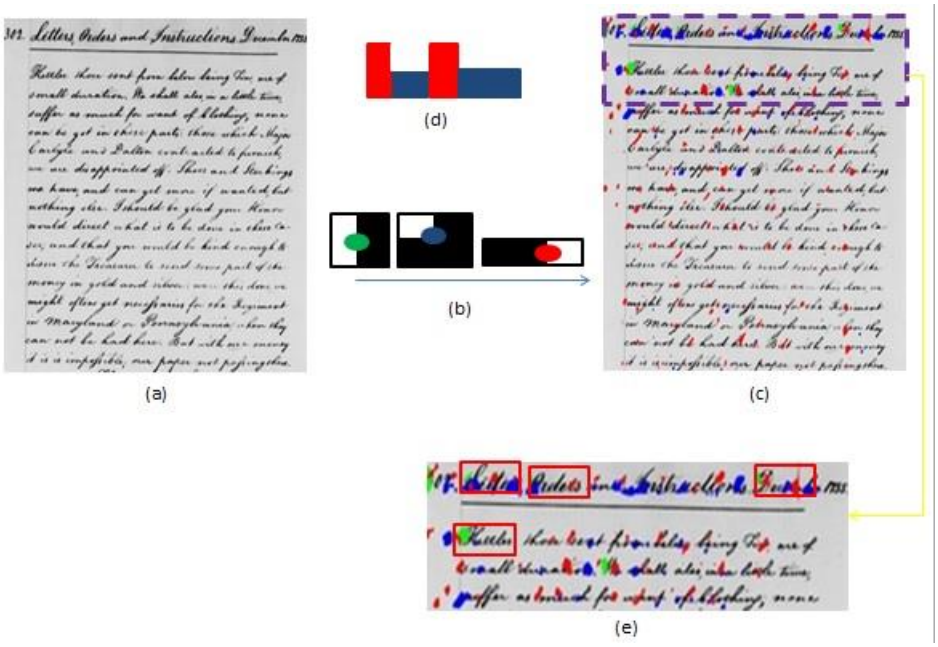


**Figure 62: A vote accumulation example (a) The RGB input document image. (b) The different GHFs applied on the RGB document. Here we illustrate that every response obtained by the application of any GHF is restored in the middle center between the black and white adjacent rectangles. (c) The different responses obtained by applying these 3 GHFs. The green regions indicate the responses generated by the first pattern. The blue regions illustrate the location of responses generated by the second pattern. Finally, the red regions represent the response generated by the third pattern. (d) The textual region that are looking for. (e) A zoomed part of the resulted image (c) where the CWs are bounded by boxes.**

Generally, all the responses generated by the application of the different GHFs in our approach are restored in the center between adjacent white and black rectangles. These responses are accumulated at the neighborhood of the beginning of each candidate words.

Thus, as we illustrate is figure 62 (e), the spatial region that contains the three colors representing each type of responses is considered as a potential candidate region. However, if not, the possibility of the existence of other candidates is not doubted. Sometimes, the process doesn't well respond to some candidate words due to the quality of the document, or other constraints that might be encountered.

The accumulation of the various votes is illustrated in figure 63.

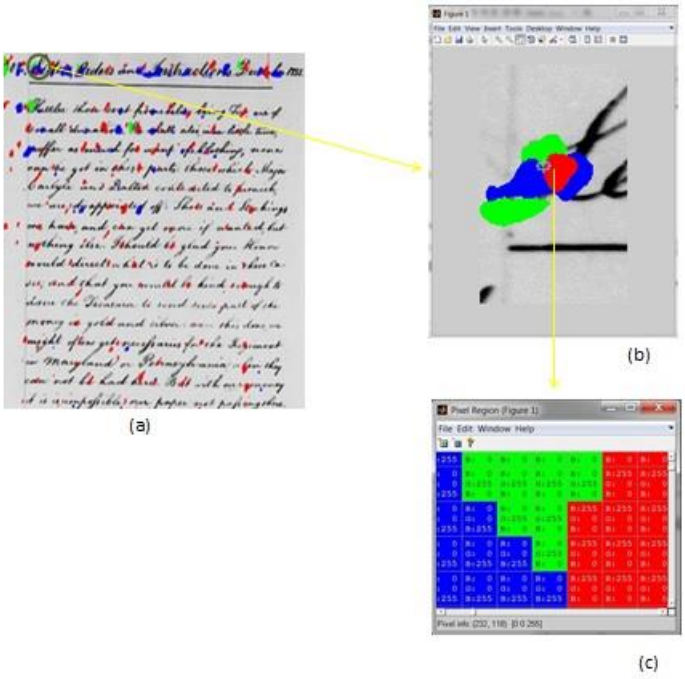


**Figure 63: An example of the accumulation of the various votes characterizing generated features after applying GHFs. At the beginning of the word letters, that is the string query of this example, various connected pixels are colored by the red, grey, and blue which represent the three features generated by applying the patterns (Figure 62(b)), so this spatial location indicates the location where the accumulation process has been done.**

Hence, every region that contains an accumulation of all types of generated features is considered as the beginning of candidate words. This helps our framework to limit bounding boxes surrounding the candidate words. Indeed, taking into consideration the information brought by the look up table as the height of a letter and the number of letters constituting the query, bounding box for each candidate region is drawn (figure 62 (e)). From the selected maximums, we built on the right of these points boxes where their sizes are in relation of the information derived from the coding process and the INT table (sub-section 3.1 and sub-section 3.2).

The global filtering step generates all potential candidate regions that are more or less similar to the searched query. The number of these candidates has still to be more reduced in order to obtain more accurate results. To refine the results generated from the global filtering step, we are going to introduce a local filtering step in our coarse-to-fine word spotting proposed approach.

## 3.4. The refining filtering module

This step is considered as a local study in our approach that allows checking if the query is present in the document at this place or not. This module is a focusing process on the CWs. The global filter-

ing results in a large number of CWs corresponding to the global shape of the query. This number is greater than the real number of the occurrences of the query. The second phase in our approach consists of a refining filtering. For that, we are going to change the observation level. The candidates have approximately the same size (number of characters) as the query. Thus, we will study all the selected words and bring more confidence to words that are most alike. The comparison and classification steps are made by the representation of selected word images by a sequence of features. Yet, we have chosen the vertical projection of the image at grey scale level that is presented in (T. Rath, S. Kane, A. Lehman, E. Partridge, R. Manmatha 2002). A hierarchic ascendant classification is then performed and which is based on the DTW matching distance computed between features sequences of bounding boxes associated with each CW.

In our context, due to the fact that we do not have any *a priori* knowledge about the common characteristics between the query and the obtained candidate words, we have chosen that the classification process will generate two classes. We assume that the class containing greater number of candidates is the class that contains more candidates similar to the query. Finally, we will obtain less candidate query occurrences existing in the processed document images and this result allows increasing the precision rate of our approach.

The flowchart of the refining step is illustrated in figure 64.


**Figure 64: The refining filtering process**

This filtering step is based on a matching process between the candidate regions. In order to choose the best matching process at word level, we have referred our choice to the work of (T. Rath et al. 2002). In this work, several matching techniques at word level have been reviewed. The Dynamic Time Warping technique was the most performing technique. Because of that, we choose to apply this matching technique within our approach. Besides, in the same work of (T. Rath, S. Kane, A. Lehman, E. Partridge, R. Manmatha 2002), many features have been evaluated on binary and grey images. As we work on grey images, we only choose the vertical projection features as the other features proposed in the same work have to be performed on a binarized image. Yet, we think that the vertical projection features that are extracted from the candidate words and which will be matched by using the DTW algorithm are pretty much enough to obtain an accurate classification.

### 3.4.1. Vertical projection

It represents the sum of the intensity values in each pixel column of the grey candidate word image. The generated feature sequence is normalized between 0 and 1 by dividing each sequence element by 255 times the height of the CW.

The next figure (figure 65) illustrates the normalized vertical projection curve of a word image of size 68×472 pixels.

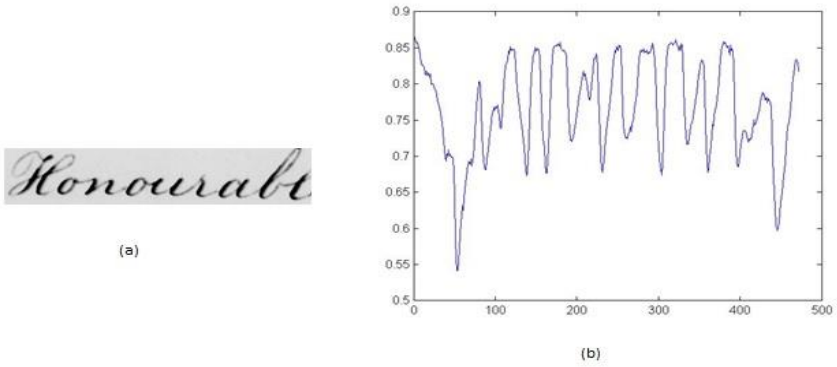


**Figure 65: Normalized vertical projection. (a) The word image (b) the curve representing the vertical projection of the word "Honourable".**

### 3.4.2. Dynamic Time Warping (DTW)

The Dynamic Time Warping technique is a dynamic algorithm that permits measuring correspondences between two different time series and computing an accumulative matching distance using their correspondences. This algorithm allows finding a local minimum matching distance measurement between two aligned time series by using the best warping path. A warping path is defined from the lower right to the upper left of a DTW matrix computed by aligning corresponding elements or samples in the given two time series. The matching distance between two aligned elements is computed by a local distance measure. Indeed, the cumulative cost of aligning all corresponding samples along the path constitutes the global matching distance between two times series. Thus, the DTW permits restoring correspondences between samples by finding the warping path with minimum cost. A detailed description and discussion of the DTW is done in (D. Sankoff, J. Kruskal 1999) and (E. J. Keogh, M. J. Pazzani 2001).

Now, let's assume that we have two candidate word images of respective widths m and n. The two vertical projections of these two candidates are represented by two numerical sequences that are $X=(x_1, x_2,.., x_m)$ and $Y=(y_1, y_2,\ldots, y_n)$. In order to determine the DTW distance between X and Y, we built a matrix D of m×n where each matrix element (i,j) corresponds to the cost alignment between the points $x_i$ and $y_j$. The computation of each entry of the D matrix is computed by the following formula:

$$
\begin{aligned}
&D(1,1) = d(x_1, y_1) \\
&\quad for\ i = 2\ to\ m \\
&\qquad D(i,1) = D(i-1) + d(x_i, y_1) \\
&\quad for\ \ j = 2\ to\ n \\
&\qquad D(1, j) = D(1, j-1) + d(x_1, y_j) \\
&for\ i = 2\ to\ m \\
&\quad for\ \ j = 2\ to\ n \\
&\qquad D(i, j) = \min\begin{Bmatrix} D(i, j-1) \\ D(i-1, j) \\ D(i-1, j-1) \end{Bmatrix} \quad + \ d(x_i, y_j)
\end{aligned}
\qquad (26)
$$

$d(x_i, y_j)$ defines the distance between points in the couple $x_i$ and $y_j$. The used distance is a dissimilarity distance defined as the squared Euclidean distance:

$$d(x_i, y_j) = (x_i - y_j)^2 \tag{27}$$

Indeed, the first component of the equation 28 (see formula 26) is defined by three values which represent a local continuity constraint.

$$\begin{pmatrix} D(i, j-1) \\ D(i-1, j) \\ D(i-1, j-1) \end{pmatrix} \tag{28}$$

This local continuity constraint is a neighborhood relationship constraint ensuring that all samples in the two input sequences belong to the warping path.

Indeed, the warping path is computed by backtracking the minimum cost path from the entry D(m,n) to the entry D(1,1). Hence, the accumulated distance along the warping path is stored in D(m,n).

Yet, the matching distance between two candidate words is calculated by comparing their generated vertical projection feature sequences using the DTW algorithm. Here, we will have two cases:

- If the matching distance between the two candidates is large, then, the two candidates are not very much alike.
- If the matching distance between the two candidates is small, then, the two candidates are pretty much similar.

As the DTW matching process is performed on several candidate words, then we will obtain various matching distances. In fact, let's assume that we have x Candidate Words (CWs) CWs=$CW_1$, $CW_2$, …$CW_x$. In order to store the matching distance between each couple of the false positives, we construct a diagonal symmetric matrix DC of size x×x where the ($i^{th}$, $j^{th}$) element of the matrix contains the matching distance between two candidate words ($CW_i$, $CW_j$). Each matrix element (i,j) where i=j corresponds to the matching distance between two same candidate words, thus it is equal to 0. The DC matrix is illustrated in the following manner:

$$DC = \begin{bmatrix} 0 & v_{(CW_1, CW_2)} & \cdots & \cdots & v_{(CW_1, CW_x)} \\ v_{(CW_2, CW_1)} & \ddots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \ddots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \ddots & v_{(CW_{x-1}, CW_x)} \\ v_{(CW_x, CW_1)} & \cdots & \cdots & v_{(CW_x, CW_{x-1})} & 0 \end{bmatrix} \tag{29}$$

Where $v(CW_i, CW_j)$ represents the matching distances between the two candidate words $CW_i$ and $CW_j$.

As the objective of the refining filtering step is to increase the precision rate by decreasing the number of false positives, the idea is to cluster these candidate words into classes depending on the matching distance similarity between them. As long as we deal with false positives or candidate words that are more or less similar to the query word, then the class with few elements will be deleted as these elements have a high variability compared to other elements of the other class.

### 3.4.3. A Hierarchical Ascendant Classification (HAC)

By generating all matching distances amongst the candidate words, we would like to build the partition of the candidate words into two homogenous clusters which are different one from the other. Thus, two candidate words belonging to the same cluster are somehow close to each other and two candidate words belonging to different classes are somehow far from each other. In this context, we perform a hierarchical ascendant classification algorithm (HAC) where the threshold is chosen on order to cut the hierarchical tree in 2 parts. This threshold permits obtaining only two clusters for all candidates. An illustration of the HAC process for clustering for instance 10 CWs into two classes depending on their DTW matching similarity is shown in the dendrogram of figure 66.

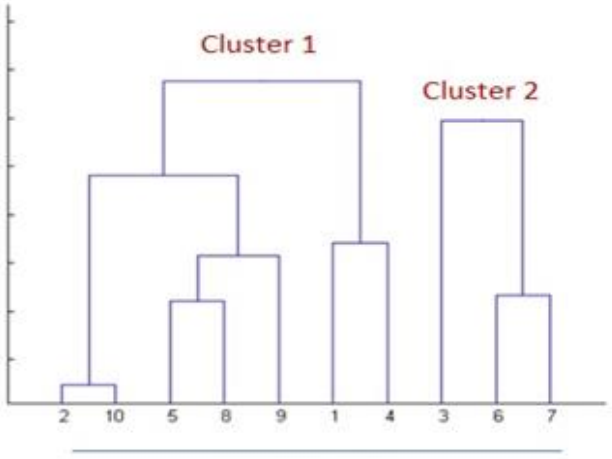


**Figure 66: The HAC process for grouping homogenous candidate words.**

The hierarchical clustering process is based on the DC matrix (sub-section 3.4.2) as it contains the different matching distances between candidate words. To apply the HAC process, we have to transform the distance matrix into a similarity matrix. The values x in DC matrix are then replaced by $1-\frac{x}{M}$ where M represents the maximum of the overall matching distances in the DC matrix.

Consequently, we obtain a new matrix known as DC1 matrix and which is obtained by formula 30.

$$DC_1 = \begin{bmatrix} 1 & 1-\frac{v_{(CW_1,CW_2)}}{M} & \cdots & \cdots & 1-\frac{v_{(CW_1,CW_x)}}{M} \\ 1-\frac{v_{(CW_2,CW_1)}}{M} & \ddots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \ddots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \ddots & 1-\frac{v_{(CW_{x-1},CW_x)}}{M} \\ 1-\frac{v_{(CW_x,CW_1)}}{M} & \cdots & \cdots & 1-\frac{v_{(CW_x,CW_{x-1})}}{M} & 1 \end{bmatrix} \quad (30)$$

Finally, after clustering the candidate words into two classes, we retain only the class which contains more candidates assuming that these candidates are those who are more similar to the query word taking into consideration all the previous assumptions mentioned in the previous section 3. In our context, the decision of awarding a false positive as relevant is not evident. Thus, this process is going to be kind of heuristic in our work. Yet, by eliminating the class with fewer candidates, we note that the precision rate of our word spotting approach is increased. However, the recall rate does not change.

After describing and detailing our QBS segmentation free word spotting based approach, now we are going to evaluate the method and compare the results with the state of the art results.

# 4. EXPERIMENTS

In the experiments, we have designed an experimental protocol for the evaluation of our work.

This protocol is as follows.

## 4.1. Evaluation protocol and metrics

### 4.1.1. Evaluation protocol

The proposed query-by-string analytical Word Spotting approach has been evaluated on the George Washington (GW) Database introduced in (T. Rath, R. Manmatha 2007) (A. Fischer, A. Keller, V. Frinken, H. Bunke 2012). This database consists of 20 pages of historical handwritten cursive texts written during the 18th century. It contains:

- 20 pages
- 656 text lines
- 4,894 word instances
- 1,471 word classes
- 82 letters

These documents are considered as longhand scripts that were written in black ink by two writers. Besides, this database contains a ground truth that contains transcription at line level and transcription at word level. Indeed, the main block of text is extracted from the original document images.

The manuscripts within this database are characterized by:

- Variability in writing style: even so the manuscripts are written by only 2 writers, we notice that there is an alteration within the writing styles.
- Texts with different heights.
- Discontinuous words.
- Texts with different widths.
- Variability of the quality of the ink: it may happen within the same document page.
- The body line of the script is more or less horizontal: there is no curved text.

Besides, it has also been evaluated on the database used in the keyword spotting for handwritten documents competition of ICDAR2015[10]. It has been set in the tranScriptorium project[11]. This database contains 79 document pages from the Bentham collection[12][13] written by the philosopher Jeremy Bentham (1748-1832) and by Bentham's secretarial staff. The ground truth is provided at line level in (B. Gatos, T. Causer, K. Grint, V. Romero, J. A. Sanchez, A. H. Toselli, E. Vidal 2014).

The manuscripts within this database are characterized by:

- Variability in writing style.
- Crossed-out words.
- Underlined words.
- Text with different heights.
- Text with different widths.
- Overlapped writings.
- The body line of the scripts is irregular.
- Noisy background.

Both the GW database and the Bentham database represent various writing styles and several characteristics which deliver some challenges for our word spotting approach.

### 4.1.2. Evaluation metrics

First of all, the evaluation of our word spotting approach on the GW database is processed in two levels which are:

- similarly to (T. Rath, R. Manmatha 2007), (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007), and more recently (A. P. Giotis, G. Sfikas, C. Nikou, B. Gatos 2015), we selected the same 15 words as queries. The list of these words is shown in figure 67.

[10] http://transcriptorium.eu/~icdar15kws/index.html
[11] http://transcriptorium.eu/
[12] https://www.ucl.ac.uk/library/special-collections/a-z/bentham
[13] http://digitool-b.lib.ucl.ac.uk:8881/R&?local_base=BENTHAM

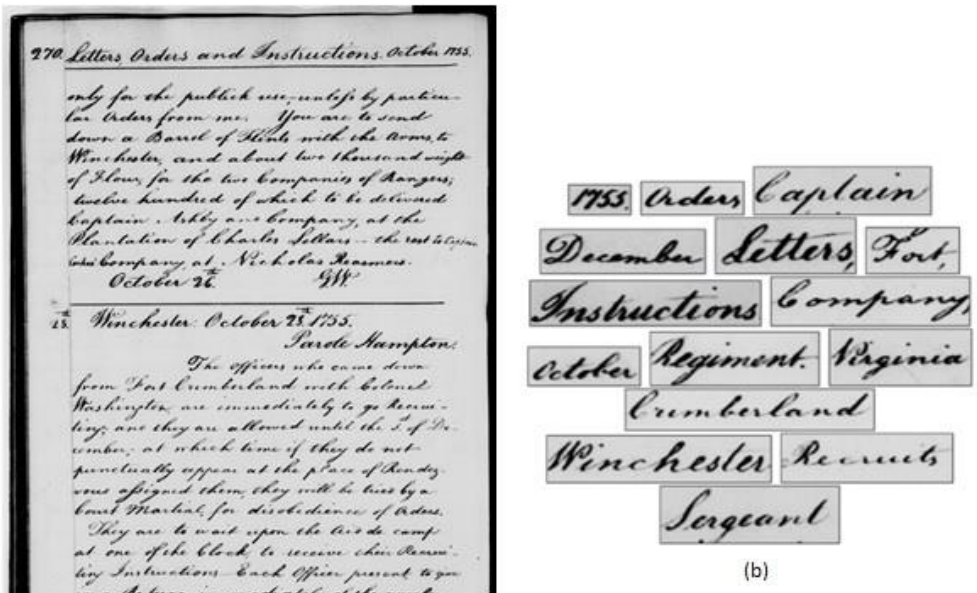


**Figure 67: A manuscript of the GW database. (a) A sample page. (b) The 15 used queries.**

However, as our approach is query-by-string based, we do not use query images as the works previously mentioned, but we type them with the keyboard. These words are the most significant in terms of semantics and occurrence frequency.

- Apart from the latter list that is composed by 15 queries, there is no other authentic query set in the literature intended for the evaluation of the GW dataset. Thus, we enlarge the set of query words from 15 to 100 queries of different lengths. Indeed, the user has the benefit to type any string queries. As an advantage of the use of a query-by-string technique, the typed string query may not exist within the manipulated document images.

Secondly, we adopt the keyword list that has been used in the query-by-string assignment in the ICDAR 2015 competition on keyword spotting for handwritten documents[14]. This list consists of 243 string queries of different lengths (6-15 characters). As previously, we may add some queries that do not even exist within the Bentham collection dataset. The figure 68 shows a sample processed document page of the Bentham dataset and some keyword query images.

[14] http://transcriptorium.eu/~icdar15kws/data.html

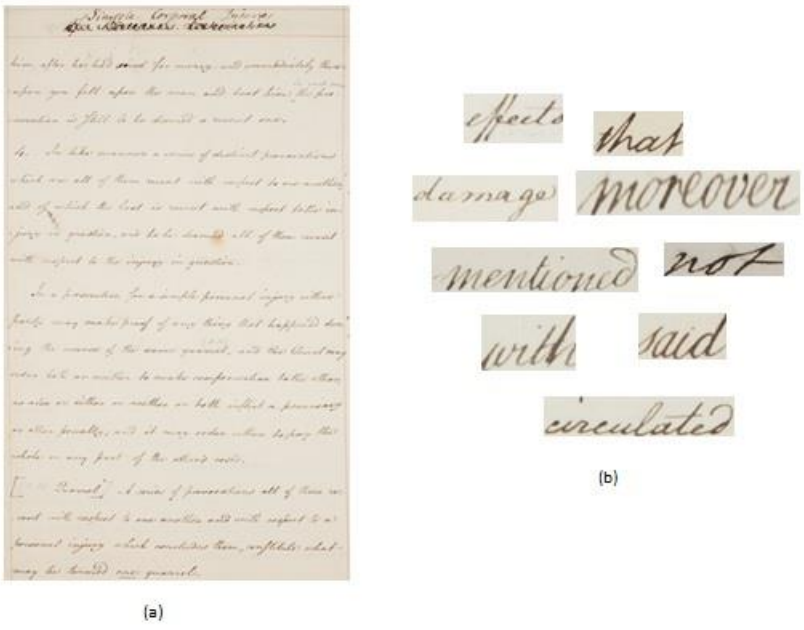


**Figure 68: A manuscript of the Bentham dataset. (a) A sample page. (b) Some keywords queries.**

Indeed, the performance measures that we used for the evaluation are Precision (P), Recall (R), and the F1 score. These metrics are defined in chapter IV sub-section 4.2.

Those performance measures are in line with the majority of the segmentation-free word spotting works.

Finally, we must mention that no words in the GW 20 benchmark are removed even words with few appearances or even stop-words as it is done in same related word spotting works. However, we also selected randomly different query words and tested them on document images that do not exist within them.

## 4.2. Qualitative and Quantitative Experiments

Now, after describing the evaluation protocol and metrics used for the evaluation, we are going to show our experimental quantitative and qualitative results.

### 4.2.1. Qualitative results

In manuscript documents, the text introduces most of the time some horizontal well-contrasted part. Then, looking at the page, text brings some horizontal contour. In the adopted dataset or most of manuscript documents, the text is handwritten in either lowercase or uppercase and the text size is depending on the writing style of the writer. Hence, focusing on each document, we may remark that the size of written letters is not the same. Besides, since the scripts of the GW dataset is written in black ink, the quality of the writing is not always good. Some characters may not be well written. The degrada-

tion of the quality of the writing affects the results generated by applying the generalized Haar-like patterns on the manuscript. In fact, these patterns are defined by connected black and white rectangles and they compute the difference between the sum of pixel intensities within white rectangle and pixel intensities within black rectangle. Thus, the generated features will not be accurate if the ink contained in the black rectangle has a poor quality. This increases the number of the false negative which decreases the specificity of the proposed approach.

Let's consider that we want to search the string query "Letters". This query is typed with the keyboard. Then, by performing the "rectangular-shape" modeling technique, this query is coded into a look up table that contains the different rectangular shapes representing it and the index numeric table containing the indexes of each element of the query. The string query is formed by 7 characters: one uppercase letter followed by a lowercase letter, followed by two lowercase letters with ascenders, followed by 3 successive lowercase letters without ascenders or descenders. The constructed generalized Haar-like filters are shown in figure 69 (b).The spotted candidate words are figured in figure 69 (c) bounded by rectangular boxes. The height of these boxes is set as the estimated height of the text line (sub-section 3.2) and the width is fixed as the length of the query multiplied by the estimated height.

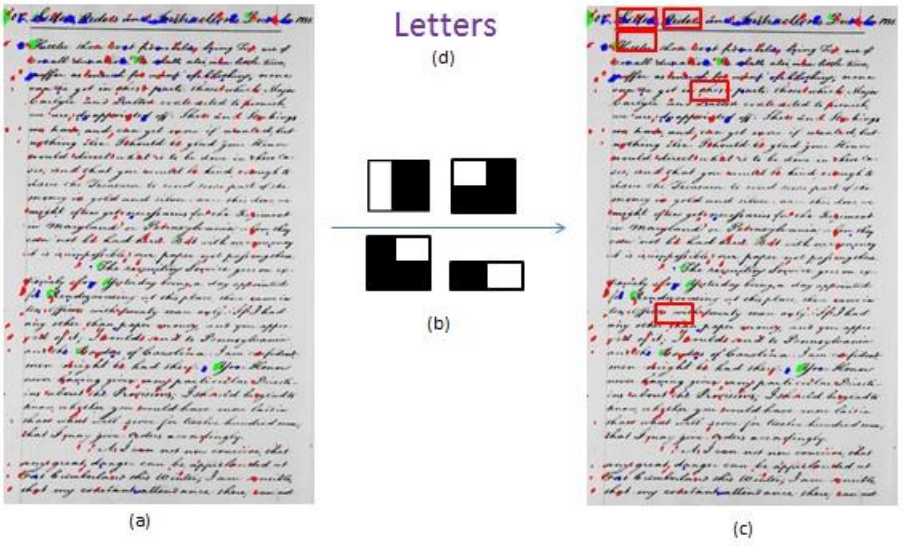


**Figure 69: The spotted CWs of the string query "Letters". The spotted words within the bounded boxes are similar to the query in terms of shape and characteristics.**

We note that our proposed method presents some false positives in the returned responses. We can notice that those false positives are highly similar to the query in terms of shape. For instance, we obtain results such as: "Orders" and "Kettles". These false positives have approximately the same length as the query "Letters" and may represent the same global shape. These three words begin with an uppercase letter followed by a lowercase letter without ascender or descender and end with a consecutive set of lowercase letters without ascenders or descenders and a blank space. Besides, there are two lowercase letters with ascenders in the middle of the query "Letters", one lowercase letter with ascender in the middle of the spotted words "Orders" or "Borders", and three lowercase letters with ascenders in the middle of the third spotted words "Kettles". It is expected to obtain such results because we apply a word spotting approach that does not recognize words but their global aspect.

Our proposed framework is a query-by-string based approach. The advantage of such type of approaches is that we can select queries that do not exist within the manipulated documents. This could not be done in the query-by-image based approaches. Thus, we can take "December" a string query

and we run our algorithm on a document image that does not contain the word December. The results are illustrated in figure 70.

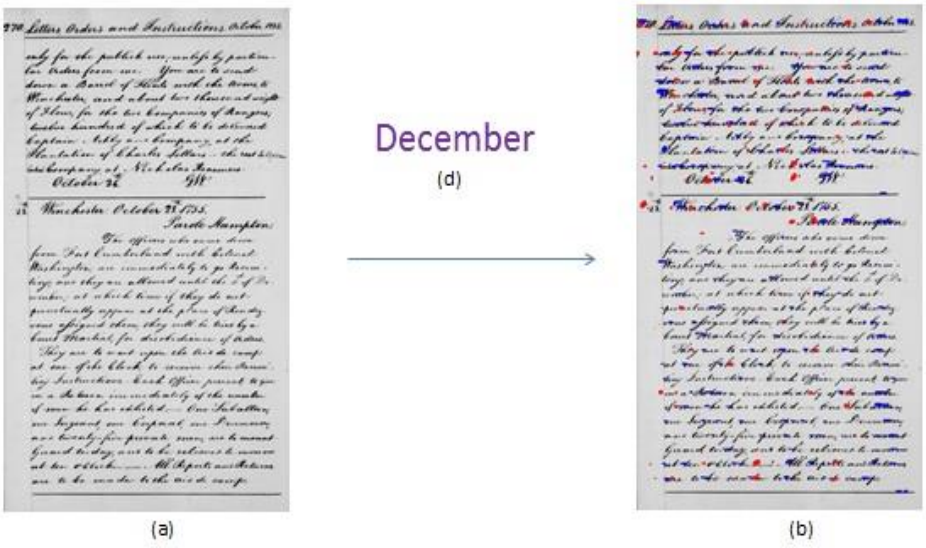


**Figure 70: An example illustrates the advantage of using a query-by-string technique that is: the query may not exist within the manipulated document page.**

We notice that there are no spotted candidate words in this case. Thus, the tested document image does not contain the word “December” and the framework takes into consideration the advantage of the QBS technique.

In addition, we mentioned earlier that the quality of the Ink may affect the word spotting process. More precisely, it influences the features generated by applying the different GHFs. To illustrate this constraint, we choose the same string query as the previous example and we choose a manuscript image that contains various occurrences of the query word. We made sure that the quality of the black ink, which these occurrences are written with, is degenerated. In figure 71, we notice that we obtain a false negative. There is a word “December” within the tested document that is not spotted. This is due to the poor quality of the ink and the overlapping of the beginning of “December” with the end of its neighboring words. In such cases, where the quality is poor, the application of GHFs in these locations failed.


**Figure 71: An example of the degradation of the quality of the black ink. (d) Represents the query**

We present, in figure 72 and figure 73, two examples of some qualitative results for two given string queries for the GW dataset. These false positives are in most of the cases pretty much similar to the given typed query in terms of shape and lengths.

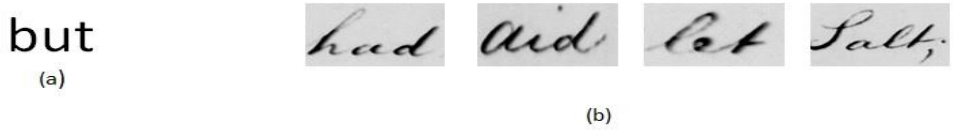


**Figure 72: An example of spotted word for a string query searched in the GW dataset. (a) The typed query. (b) False positives.**

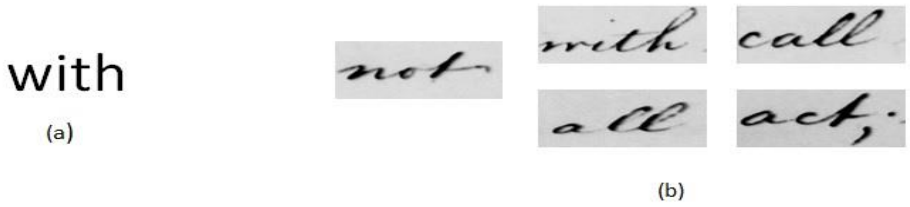


**Figure 73: An example of spotted word for a string query searched in the GW dataset. (a) The typed query. (b) False positives.**

Furthermore, the manipulated document pages of Bentham dataset present few constraints that may bring out some difficulties in the process of the proposed word spotting approach.

We are going to show a bunch of qualitative results obtained after applying GFH's on document pages representing some constraints such as: (i) Overlapped writing (ii) Background with noise (iii) crossed out words. We illustrate these results in order to study the stability of applying GHFs on such particular document pages.

By applying a GHF which allows detecting both the beginning of words and letters with ascenders or descenders on a document where an overlapped writing exists (figure 74 a), the responses obtained in the area, where the writing is not well spaced, are overlapped. Thus, the discrimination between

candidate words is not evident. In fact, as we mentioned above (chapter III), Haar-like features are defined by the intensity difference between the black and white areas of the applied GHF, so these features will not be very representative in the case where we face overlapping words or text lines (figure 74 b). As a conclusion, our approach will somehow come across some problems in spotting string queries that some of their occurrences are overlapped with writing.

Besides, by applying different GHFs (figure 75 (b)) on a document image with noisy background (figure 75 a), the noisy background responds accurately to these patterns (figure 75 c).Yet, the accumulation vote process when obtaining different responses from different GHFs will be discriminative.

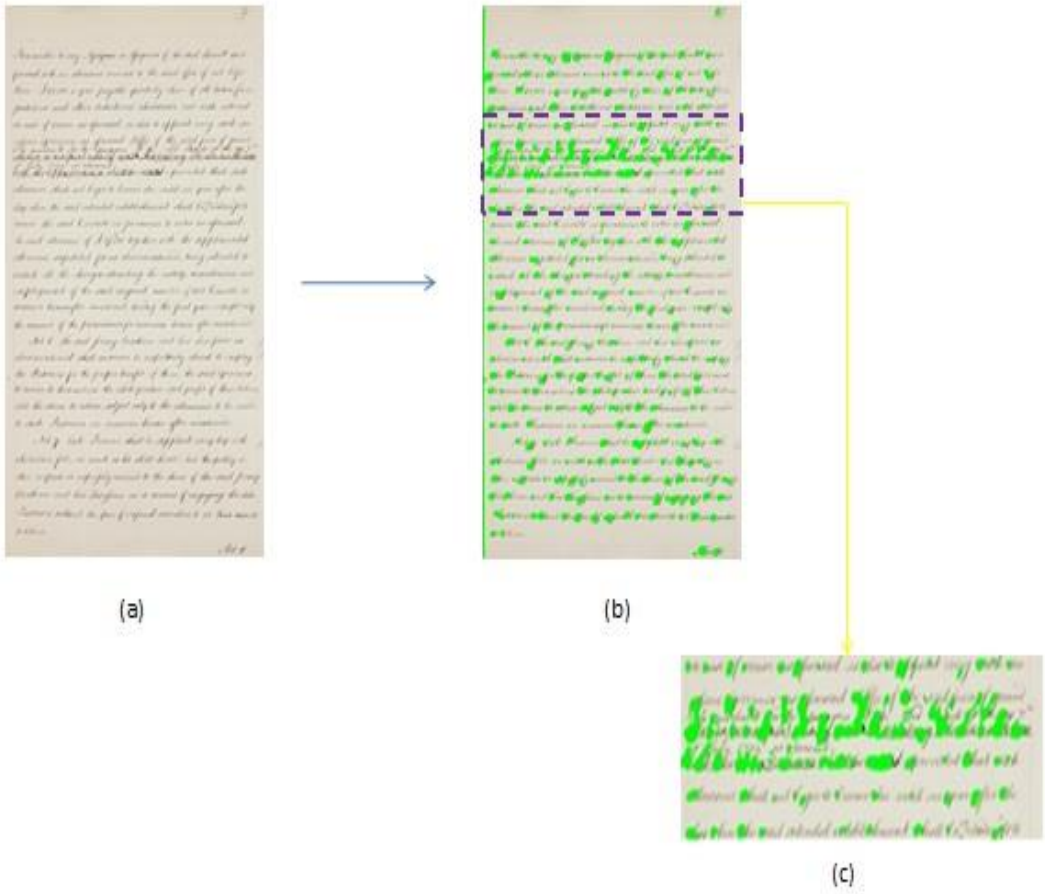


**Figure 74: An example of overlapped writing in a document page from Bentham Dataset. (a) The original document page. (b) The green responses indicate the beginning of words and the letters with descenders or ascenders within the tested document image. (c) A zoomed region highlight generated responses of overlapped textual components.**

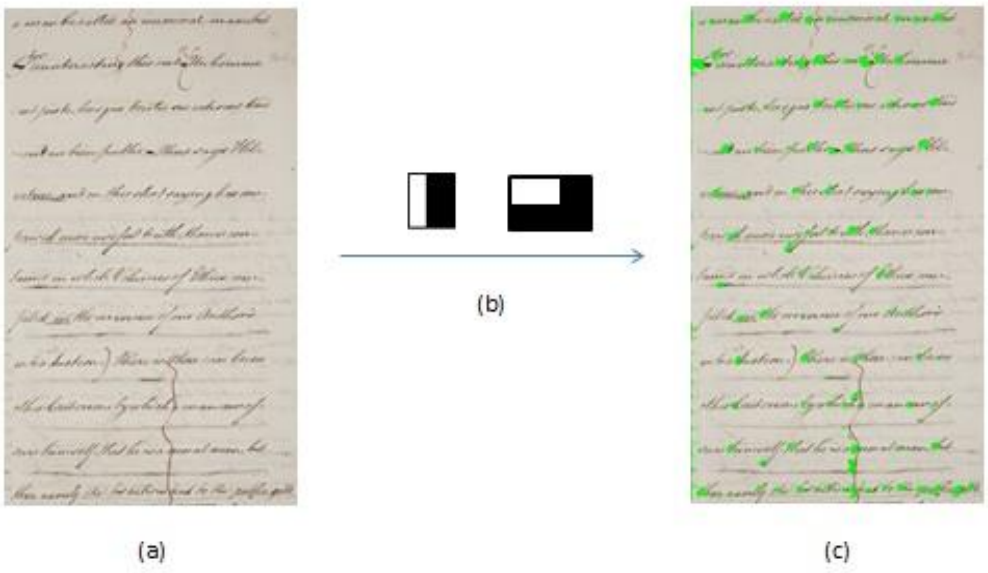


**Figure 75: An example of noisy background of document page from Bentham Dataset. (a) The original document page. (b) The green responses indicate the beginning of words and the letters with descenders or ascenders within the tested document image.**

Besides, the same observation statements as those mentioned in the case of document with overlapped writing are pointed out when we deal with crossed text lines (figure 76).

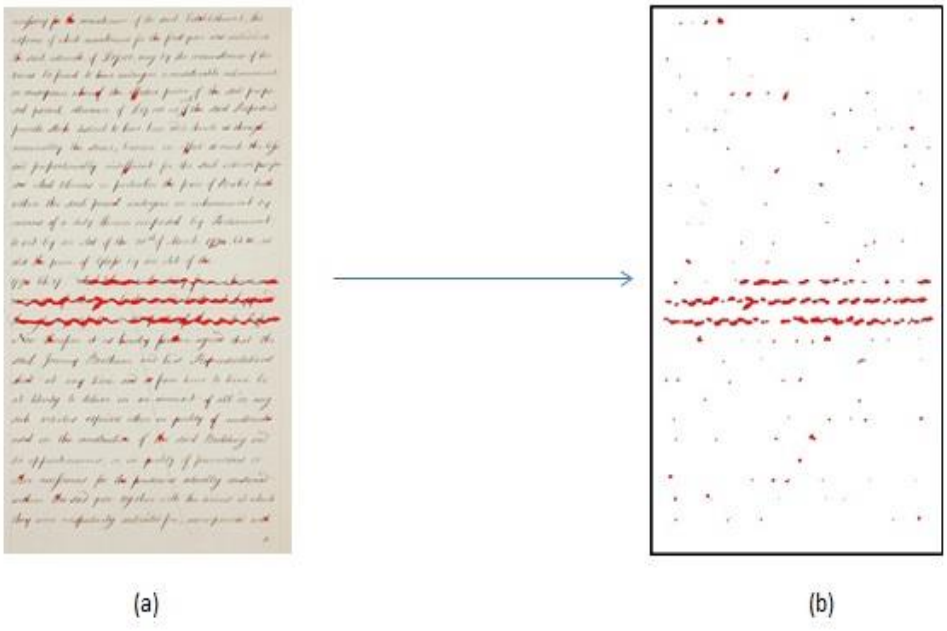


**Figure 76: An example of a document page from Bentham Dataset where some text lines are crossed out. (a) The crossed text lines respond too much to the applied GHF. (b) We highlight only the red responses without textual components. For the Applied GHF, we used the first one in figure 75(b).**

We present, in figure 77 and figure 78, two examples of some qualitative results for two given string queries for the Bentham dataset. These false positives are in most of the cases pretty much simi-

lar to the given typed query in terms of shape and lengths. Besides, the different writing styles within the document of the Bentham dataset influence the results. This has an impact factor on the generated Haar-like features tends to increase the number of false positives.


**Figure 77: An example of spotted word for a string query searched in the Bentham dataset. (a) The typed query. (b) False positives.**

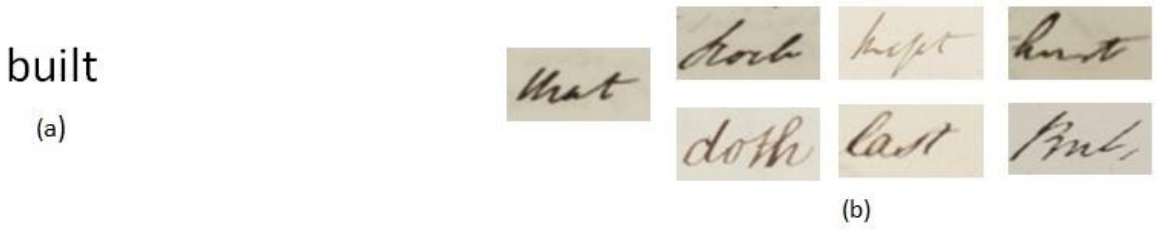


**Figure 78: An example of spotted word for a string query searched in the Bentham dataset. (a) The typed query. (b) False positives.**

In this section, different qualitative results on both GW and Bentham Datasets were presented. Besides, some constraints which perturb the proposed process of word spotting are highlighted. So now, we are going to quantitatively evaluate the performance of our word spotting approach.

### 4.2.2. Quantitative results

We would like to mention that there is no standard criterion and there is also a lack of standard databases on which to compare our evaluations with other works. In fact, word spotting techniques have been experimented on several types of datasets with different number of tests and queries. All of that makes the common measures for evaluating the quality of the obtained results not adequate to our context. Besides, the aims of the different applications may differ. This is considered as a real problem in the word spotting domain that must to be solved in near future (further information are mentioned in chapter II).

For the evaluation on the GW dataset, we compare our results with two other word spotting approaches presented in (T. Rath, R. Manmatha 2007) and (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007). We use all the 15 queries that have been used in these latter works. The list of the words used is shown in figure 67(b)). However, as our approach is query-by-string based, we do not use query images but we type them with the keyboard. Indeed, we will mention the evaluation metric for using all the occurrences of words existing in manipulated document images.

Before we give the different quantitative result rates, we are going to remind some characteristics of the GW document collection to highlight the efficiency of our word approach. In GW database, there are two categories of words which may be a thin word class and a large bold class. Besides, the writing style is variable between each processed document. Indeed, the same character may have different sizes in the same document image. We will show by mentioning the recall, precision, and F-score rates, that our approach is able to handle these variations.

We take advantage of this section to properly describe and illustrate some results obtained in both the global and refining phases. Thus, we are going to compare the results of spotting the 15 queries (figure 67(b)) before and after the refining step.

Indeed, considering the different constraints mentioned above, for each query searched within all the manuscript of the GW database, we achieve throughout the global filtering phase an excellent recall rate of 96.6 % (figure 79), whereas Rath obtained a recall rate of 90.72% (T. Rath, R. Manmatha 2007) and Leydier only obtained a recall rate of 74.2% (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007) . Of course, this result is not significant if it is not related to the precision rate.

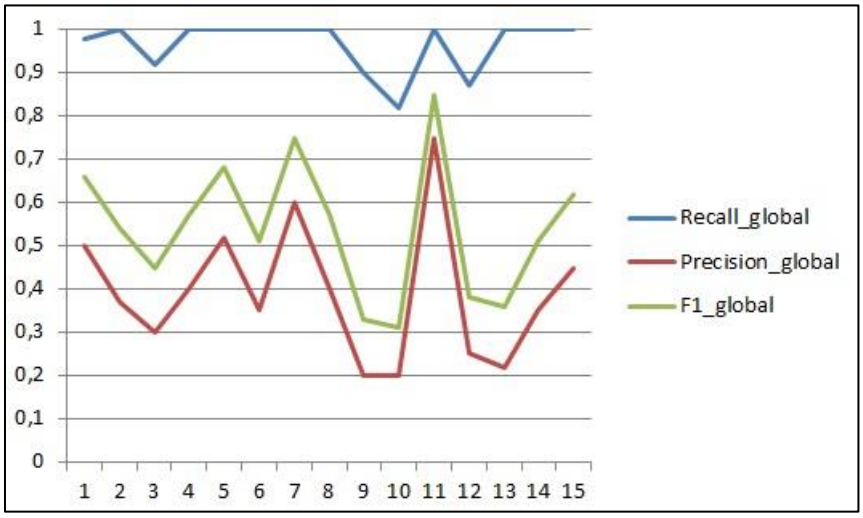


**Figure 79: The recall, precision, and F1 rates of the 15 used queries throughout the global filtering stage.**

For the precision, the two works that we are comparing our experiments with, do not mention accurately either their precision or F1 score in spotting the different queries. The average of precision rate of spotting the all 15 queries is of 39%, and the F1 score is of 55.81%. The curves in figure 79 indicate the recall and precision rate for each query.

The low precision rate is due to the false positives generated by our approach. These false positives share several characteristics (i.e. number of characters, etc.) with the keyword query. They are similar or relevant to the query (i.e. figure 80).We try to group homogenous false positives in term of their DTW matching distances of their vertical projections into two clusters. We assume that the cluster containing several alike candidates is the relevant collection to the query. This conveys that the elements of this collection are relevant to the query. The false positives contained in the second cluster will be considered as negative and will be rejected. Here, we do not display all the true positives in order to simplify the demonstration of the HAC process.

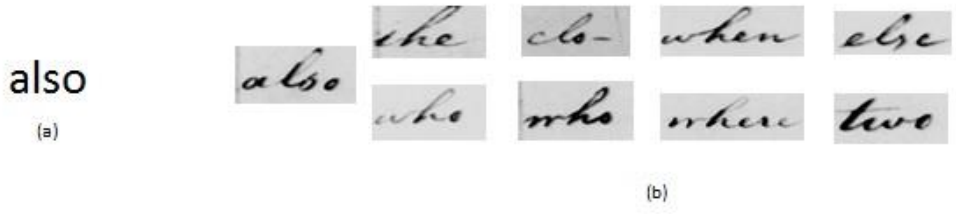


**Figure 80: An example of CWs for a given typed query. (a) The query. (b) The CWs of the query.**

The classification process is accomplished by a hierarchical ascendant clustering process where the cutting level is set as 2. This technique associates candidates presenting low variability between their DTW matching distances of their vertical projections into two classes. The figure 81 illustrates the hierarchical classification process of the query word “also”.

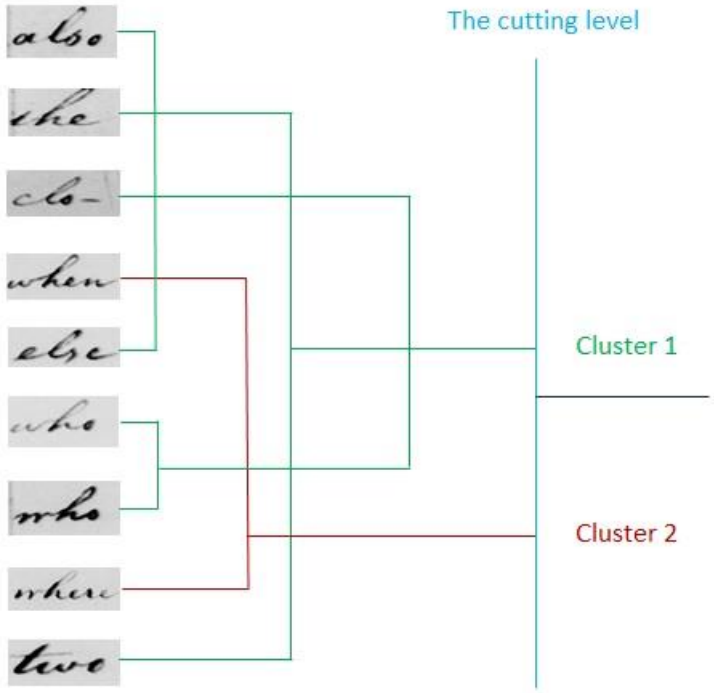


**Figure 81: The HAC process for a given query "also".**

By eliminating the candidates that are considered as negative, the number of false positives decreases. Then, the precision rate increases. Thus, a refining phase based on the HAC process will gradually increase the precision rate obtained by the global filtering phase.

In order to show the progression in term of precision rate, we illustrate in figure 82 the new precision rates of each 15 queries used previously. The new average of precision rate of spotting the all 15 queries is of 61%. The recall rate is constant. Then, the F1 score is of 74.77% (figure 82).

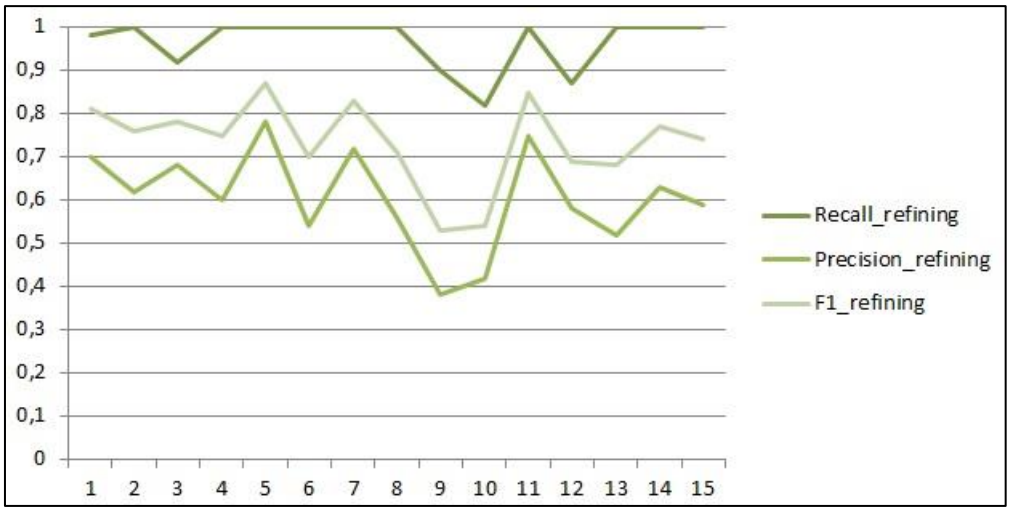


**Figure 82: The recall, precision, and F1 rates of the 15 used queries throughout the refining filtering stage.**

To see clearly the difference between the recall and precision values obtained by performing or not a filtering process, we overlap the two figures (figure 79 and figure 82) into one figure (figure 83).

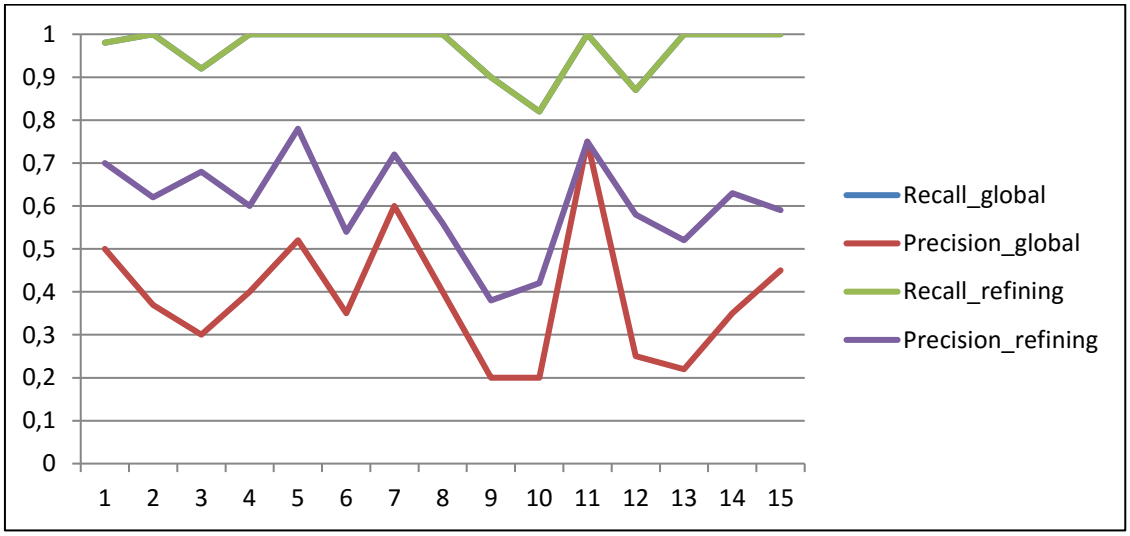


**Figure 83: The recall and precision evaluation of the WS process within the GW database by overlapping figure 79 and figure 82.**

Enlarging the string query list from 15 to 100, we do not restrict the searching process to some class of words, such as words containing at least 3 characters or words appearing at least 10 times within manipulated documents. Thus, we hazardously type 100 string queries. In this way, some queries may exist or not within the document pages. Consequently, we obtain a recall rate of 95.6%, a precision rate of 78.2% and a F1 score of 86.02%.

However, we mention that recent works of word spotting have used other evaluation metrics on their experimental evaluation. These metrics are such, mAP (defined in the two following formulas), and R-precision that indicate the precision of the $R^{th}$ position of the list of results where R corresponds to the relevant words for the queries (P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Liadõs, A. Fornés 2014b).

Indeed, for a set of queries, the Mean Average Precision is the mean of the average precision for each query. It is defines as:

$$mAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q} \tag{31}$$

Where Q is the number of queries and AveP is computed by:

$$AveP = \frac{\sum_{i=1}^{K} (P_i \times rel_i)}{numberofrelevantwords} \tag{32}$$

Here, the $rel_i$ is an indicator function which is equal to 1 if the candidate i is a relevant word or to 0 otherwise and K the number of relevant words.

These evaluation metrics are particularly used in the QBE based approaches where we deal with different image queries corresponding to the same query. For instance, in the work of (P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Liadõs, A. Fornés 2014b), the authors selected 1847 queries that correspond to 68 different words for the GW dataset. According to that, it is necessary to compute the mean average precision for each spotted queries to evaluate the spotting process. Consequently, similar approaches have to indicate if each item in the spotted ranked list is a relevant one according to the query or not. The relevance of each item allows computing the mAP of the system. Recently, most of the words spotting approaches evaluated on the George Washington (GW) dataset are based on the QBE problem. In addition to the learning free word spotting approach based on the graph representation in (P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Liadõs, A. Fornés 2014b) with mAP rate of 0.175, we also mention the work of (T. Rath, R. Manmatha 2007) that is based on the DTW technique and where the obtained mAP is of 0.169. Besides, (D. Fernãndez, P. Riba, A. Fornés, J. Lladõs 2014) have described a BoVW based approach, a Loci feature representation based approach, and a graph based approach and where the mAP is of 0.422, 0.072, and 0.028 respectively. We conclude from these results that the use of the Loci features on GW database brought out better results in term of mAP rate other than the use of the DTW technique or even a BoVW and graph representation.

Furthermore, our QBS based approach overcomes a problem cited by (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007) where their work is able only to retrieve words that are only written in the same typographical style as the query, staying in the same class. Otherwise, our WS framework is able to retrieve all the occurrences whatever its style class. Moreover, it is able to achieve good rates whatever the length of the query is. Some approaches such as the one proposed in (M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós 2011) is highly related to the length of queried words.

The proposed QBS free learning analytical word spotting approach has been also evaluated on the Bentham dataset described in sub-section 4.1.1. The manuscript document pages within this dataset possess different properties that do not exist in the GW document pages. The crossed line texts, the noisy background, the variability of writing styles, and the overlapped writings challenge the spotting framework. We illustrate the behavior of our WS framework throughout qualitative results in sub-section 4.2.1. For the quantitative evaluation, we have done the experiments on the Bentham dataset

provided in the ICDAR 2015 competition on Keyword Spotting for handwritten documents. The flowchart and the various tracks and assignments proposed in this competition are illustrated in figure 84.

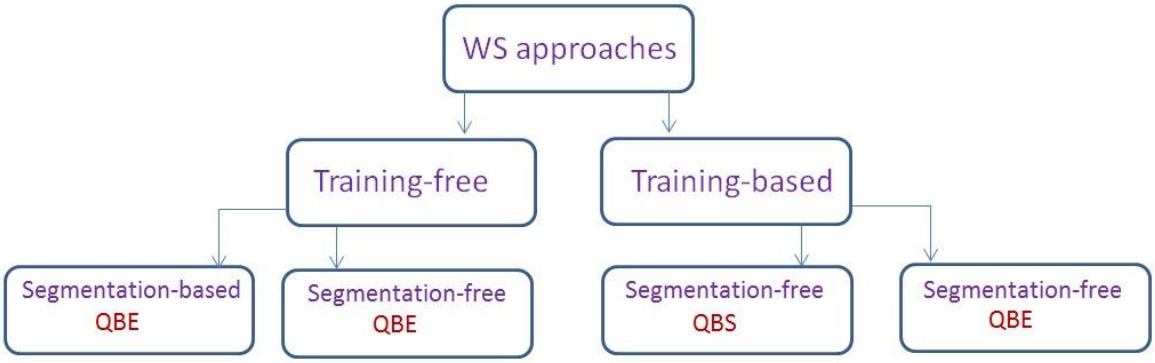


**Figure 84: The taxonomy of the different tracks proposed in WS ICDAR 2015 competition.**

According to this taxonomy, the QBE problem is used in the two tracks representing the training free and training based word spotting approaches. For the training free track, the QBE is used either in segmentation based methods or segmentation free based methods. Otherwise, the QBE is used in segmentation free based methods for training based approaches. However, the QBE problem is used only in segmentation free training based word spotting approaches. Due to this classification presented in this competition, we cannot adopt either of this categories to compare our work with because the specification of our proposed approach. The proposed word spotting approach is a free learning approach that does not rely on any segmentation process and the main specification is the use of the QBS problem. Due to that, we may mention that our WS approach is designed for a track for training free with segmentation free using QBS problem. We may recommend that this may be considered as a new assignment for future WS competition.

The WS approaches proposed in the ICDAR 2015 WS competition use the mAP metric as one of the different evaluation metrics proposed in the competition. However, as we mentioned earlier in the GW quantitative evaluation, it is not possible to evaluate our WS framework with such metrics.

The evaluation query-by-string keywords list contains words written in uppercase. We generate another list containing the same keywords but written in lowercase. The list is composed of 243 string queries of different lengths that are between 6 and 15 characters. These keywords are cited at least 4 times in the validation dataset. By spotting them within the Bentham dataset, we obtain a recall rate of 83.64%. The miss-spotted keywords are principally due to the crossed line texts and the overlapping writings. Besides, the spotting process generates various word candidates for each searched query. The application of the HAC algorithm based on the DTW matching similarities (sub-section 3.3) allows obtaining a precision rate of 58.32%. The low precision rate is caused by the different constraints of the manipulated Bentham document pages (sub-section 4.1.1). As a result, we obtain a F1-score of 68.72%. Using the original keywords list (containing keywords in uppercase), the Recurrent Neural Network (RNN) word spotting based approach proposed by the Computational Intelligence Technology Lab (CITlab) of the university of Roctock of Germany (J. Puigcerver, A. H. Toselli, E. Vidal 2015) obtained the best mAP score of 87.11%. In contrast of that, the best mAP score obtained by applying a training free segmentation based KWS approach is of 42.44% and the best mAP score obtained by applying a training free segmentation free KWS approach is of 34.3%. More information about the results of the competition are provided in (J. Puigcerver, A. H. Toselli, E. Vidal 2015). Since we propose a training free segmentation free QBS KWS approach we cannot fairly compare our quantitative results with those of the competition because we do not use uniform data and metrics.

# 5. CONCLUSION

In this chapter, we have presented a coarse-to-fine Word Spotting approach. It is a learning free approach that does not rely on any previous segmentation step. This approach is based on the QBS problem. Each string query is typed by the user through a keyboard. The proposed WS approach is based on three principal phases. First, we propose a modeling word technique that allows representing each string query by a successive sequence of rectangular shapes and an index numeric table. Second, a global filtering stage which allows obtaining different word candidates of the searched string query. This phase is based on the application of GHFs globally on the manipulated document page and a vote process. As far as we know, the use and the generalization of Haar-like features have not been applied before in any word spotting systems. The last phase of our approach is a refining filtering module that permits retaining only candidates that are pretty much similar to the query. This module is based on a HAC algorithm based on the DTW matching distance between the obtained candidates from the global module. Finally, the presented work is applied on the GW database and the Bentham database of ICDAR 2015 competition on keyword spotting for handwritten documents. These datasets consist of historical documents in English with various levels of complexity.

*"All truths are easy to understand once they are discovered; the point is to discover them"* _**Galileo Galilei**

# VI. General discussion & future works

In this chapter, we remind our conclusions and reveal the future perspectives. We review the essential contribution and aspects of this thesis.

The presented work follows two directions. The first one, described in chapter IV, disposes a technique for text and graphic separation in comics. The second one, described in chapter V, points out a learning free segmentation free word spotting framework based on the query-by-string problem for manuscript documents. The two approaches are based on human perception characteristics. Indeed, they were inspired by several characteristics of human vision such as the Preattentive processing. These characteristics guide us to introduce two multi scale approaches for two different document analysis tasks which are text extraction from comics and word spotting in manuscript document. These two approaches are based on applying generalized Haar-like filters globally on each document image whatever its type (i.e. graphic document page, historical document page, modern document page). Describing and detailing the use of such features throughout this thesis, we offer the researches of document image analysis field a new line of research that has to be more explored in future. In fact, as far as we know; this is the first clear work that uses Haar-like features in extracting manuscript text from comics and spotting string queries from historical and modern manuscript documents. Besides, the proposed two approaches overcome various constraints that are assumed in most works in the literature. The two approaches are layout segmentation free and the generalized Haar-like filters are applied globally on the image. Moreover, no binarization step of the processed document is done in order to avoid losing data that may influence the accuracy of the two frameworks. Indeed, we do not use any learning step. Thus, we avoid the process of extraction features *a priori* which will be performed automatically, taking into consideration the different characteristics of the documents, during the text extraction or word spotting process.

More specifically, we bring out various contributions for each direction. Beginning with the text and graphic separation in comics, we are able to overcome numerous assumptions cited in the literature. Indeed, our proposed approach does not rely on any assumptions of the text localization, the written color, and the writing style of the text. Besides, we do not perform any frame or speech balloons segmentation technique in order to extract the text. Ending with the word spotting in manuscript document, we enumerate these contributions. First, we propose a rectangular shape modeling technique for string queries representation. Aside from different techniques that create synthetically the query with specific fonts and sizes, we propose to represent each query with a prototype composed by adjacent rectangles and an index table. This helps in knowing the shape of characters constituting the query (i.e. character with ascender, character with descender, points, space, punctuations, etc.) and the position of each character in the string query. The adaptation of the generalized Haar-like filters is done automatically regarding the characteristics of each processed document page by an automatic height estimation technique for manuscript characters. In addition to that, we introduce a translation and an accumulation process of votes. This process permits in translating the different votes, which indicates the presence of a specific pattern in the neighborhood of the pixel, and accumulate them in order to obtain zones of interests (ZOIs). Consequently, these zone of interests correspond the presence of the query in this spatial region. For the visualization of the obtained results, we implement an automatic technique that generates a threshold value for binarization of each document. This technique has the advantage to generate an accurate threshold value for the binarization process whatever the writing style of the document is. Finally, a HAC algorithm based on the DTW distance matching of vertical projection of the ZOIs is implemented. This process allows eliminating candidate words that are not very similar to the string query and retaining only candidates that are pretty much similar to the query. This method is very promising to refine the number of the spotted words in the case when we are not able to interfere to make decisions such as if each spotted word is relevant or not to the searched query.

However, apart from these contributions, we relate some drawbacks to our work for which we tribute some future perspectives as we go along. First of all, based on a series of experiments on different types of documents, the applied GHFs are not able to either detect irregular text body line or oriented words. They respond only to more or less horizontal text areas. To overcome such problems, the adaptation and the generalization of extended Haar-like filters at different angles such as 45°, 26.5°, 78.6°, and others. This is not an obvious task as we should estimate the text orientation within each processed document image. We will concentrate our work in the future in implementing an automatic method that allows adapting the GHFs to the orientation of the text in each processed documents page. This opens other extensions for our work. As we based the proposed word spotting approach on the query-by-string problem, an extended modeling word technique for the query-by-string representations have to be developed. Besides, the proposed vote accumulation technique will fail in translating the oriented GHFs. Thus, an enhancement process should be done within this technique in order to take into consideration the text orientation. Expect for the various favorable properties of GHF regarding the context of our work, these filters are sensitive in some cases. They do not well respond when the writing is brighter than the background color. They also depend on the quality of the ink when we are dealing with historical documents. Besides, these filters are kind of sensitive to the variability of writing styles. Under these circumstances, we would like in the future to add other features apart from Haar-like features to improve the results in both text localization in comics and word spotting in manuscripts. These features have to be a tradeoff with the advantages of Haar-like features.

Going more in depth in the two proposed approaches for text extraction in comics and word spotting in manuscript documents, we look forward to propose some perspectives for each approach in order to increase their efficiencies. On one hand, the proposed method of text and graphic separation in comics overcomes various assumptions and deal with lot of constraints. Experiments show that this approach generates good results. Indeed, to increase the precision, we intend to perform with a Tesseract OCR engine that will be trained with different font and language in order to filter out regions that do not corresponds to text. We also would like to test other OCRs such as ABBYY[15]or OmniPage[16]. Likewise, we would like to develop a knowledge driven approach that allows validating or rejecting text candidates along with the use of an OCR engine. Consequently, the accuracy of localizing text within comics will be increased. Then, some related applications like extracting comic characters will be recommended to be done. On the other hand, we have used a HAC algorithm that take into consideration the DTW matching distances of vertical projections of candidate words. We have assumed that vertical projection features are pretty enough for our context. However, features such those presented in (T.M. Rath, R. Manmatha 2003) or DCT coefficients that represent the overall appearance of the interior of a word will be integrated in the clustering algorithm to more refine the results. Furthermore, we have chosen from the beginning that the proposed word spotting approach will be training free. Experiment evaluations especially on Bentham dataset from ICDAR 2015 competition on Keyword spotting for handwritten documents show that training based approaches can achieve better performance than training free approaches. Thus, integrating a learning algorithm such as a RNN or an HMM is considered as a very attentiveness direction to be explored in near future.

[15] http://france.abbyy.com/finereader/
[16] http://www.nuance.fr/for-individuals/by-product/omnipage/index.htm

*Tout obstacle renforce la détermination. Celui qui s'est fixé un but n'en change pas"* _ **Léonard De Vinci**

# VII. French summary

## Contents

# 1. RÉSUMÉ

Dans cette thèse, l'objectif est de concevoir et développer une approche originale pour le word spotting dans les documents manuscrits.

Après une étude étendue de la bibliographie concernant les approches de word spotting pour tous types de documents, on a opté pour l'utilisation des filtres pseudo-Haar pour extraire de nombreuses caractéristiques en raison de ses différents avantages. Donc, on a consacré les deux premiers chapitres de cette thèse pour l'étude de l'état de l'art et à la définition et description des ondelettes de Haar et les caractéristiques pseudo-Haar.

Avant d'appliquer les caractéristiques pseudo-Haar dans le word spotting pour les documents manuscrits, on les a appliquées pour extraire les textes dans les bandes dessinées. Cette partie est très intéressante pour savoir la stabilité de l'utilisation de ces caractéristiques dans des documents graphiques complexes. On a proposé une approche qui ne nécessite aucune phase de prétraitement (comme la détection des personnages de Bande dessinée, la détection des phylactères de la bande dessinée, etc.) et permet d'extraire automatiquement tout le texte dans ces bandes dessinées. Cette approche est évaluée sur la base de données publique eBDthèque.

Dans la troisième partie, on a présenté une approche analytique multi-échelle pour le word spotting dans les documents manuscrits. Le modèle proposé comporte deux niveaux. D'abord, un module de filtrage global permettant de définir plusieurs zones candidates répondant à la requête dans le document testé. Ensuite, l'échelle de l'observation est modifiée à un niveau plus fin afin d'affiner les résultats et de sélectionner uniquement ceux qui sont vraiment pertinents. Les contributions de cet article sont l'utilisation et l'adaptation des caractéristiques pseudo-Haar dans le word spotting. En plus, une nouvelle technique permettant de modéliser les requêtes sélectionnées par l'utilisateur à partir d'un clavier est proposée. L'approche est évaluée sur la base de données publique George Washington et la base de données Bentham.

# 2. INTRODUCTION

Dans les documents, l'extraction de l'information est habituellement effectuée par des méthodes d'analyse de texte. La performance des systèmes optiques de reconnaissance des caractères désignés aussi par l'abréviation (OCR) est encore trop faible pour convertir certaines images de documents en texte, en particulier pour la reconnaissance des manuscrits. L'OCR ne présente donc pas une solution adaptée au problème de la recherche d'information dans des bases d'images de documents en raison de ses limites dans le traitement des écritures manuscrites et dans le traitement des collections dégradées anciennes. Plus précisément, les techniques d'OCR ne sont pas assez robustes pour les écritures manuscrites dans un contexte de vocabulaire ouvert. Pour cela, le word spotting est considéré comme une alternative à l'OCR traditionnel pour différentes applications comme par exemple l'indexation de documents et l'extraction d'information dans des collections de documents numériques.

Le word spotting a pour objectif de trouver dans les images de documents les multiples occurrences d'une requête, un mot désigné par l'utilisateur. Le word spotting, sans s'attacher à retrouver les lettres des mots, utilise des méthodes d'appariement, des méthodes de mesures de similarité plus globales entre images de mots. Elles permettent ainsi de créer des indexes partiels pour le document manipulé. Donc, le word spotting facilite l'indexation et la récupération de l'information suggérée comme une requête dans des documents historiques ou modernes quand ils sont relativement complexes et dégradés.

Dans la littérature, les approches de Word Spotting ont été développées, concernant divers scripts comme des scripts latins, arabes ou grecs, etc. Ces scripts se différent les uns des autres par la nature des alphabets, le nombre de caractères, la direction de l'écriture, la forme et la cursivité entre les lettres. Ils peuvent être soit manuscrits soit imprimés.

Les approches de Word Spotting ont été divisées en différentes catégories suivant des critères différents par les chercheurs en analyse de documents. Par exemple, elles peuvent être divisées en deux catégories principales basées sur des techniques de mise en correspondance des images, certaines sont des techniques d'appariement basées images et d'autres sont basées caractéristiques (J.L Rothfeder, S. Feng, T.M. Rath. 2003). Les premières techniques contiennent des méthodes qui calculent directement les distances entre les mots à partir des pixels de l'image tels que l'appariement par templates en utilisant la corrélation. Par contre, les dernières méthodes calculent certaines caractéristiques sur les images de mots et ensuite ce sont les objets portant ces caractéristiques qui vont être appariés.

On trouve aussi d'autres classifications dans la littérature (J. Lladós, M. Rusiñol, A. Fornés, D. Fernández and A. Dutta 2012). Deux principales approches de word spotting existent selon la forme de la requête. Ces deux approches sont basées soit sur des requêtes construites par une chaîne de caractères (QBS) soit sur des requêtes basées images (QBE). Les méthodes QBS (H. Cao and V. Govindaraju. 2007) utilisent comme entrée des séquences de caractères. Elles exigent généralement une grande quantité de matériels d'apprentissage car les caractères sont appris a priori et le modèle de la requête est construit à l'exécution à partir des modèles des caractères constitutifs. Dans les méthodes basées QBE (R. Manmatha, C. Han ,E. M. Riseman 1996), la requête d'entrée est une ou plusieurs images exemples du mot requête. Le problème est alors traité comme un problème de recherche d'images par le contenu. Par conséquent, il ne nécessite pas l'apprentissage mais la collecte d'un ou plusieurs exemples du mot requête. Une autre technique populaire de catégorisation divise les méthodes de Word Spotting soit dans une catégorie basée sur la segmentation, soit dans une catégorie de méthodes sans segmentation (B. Gatos and I. Pratikakis. 2009).

# 3. CLASSIFICATION DES APPROCHES DE WORD SPOTTING

Dans cette section, nous proposons une classification des approches de word spotting qui résulte en deux catégories:

- Les techniques d'analyse globale ou holistique: techniques sans segmentation, elles traitent une image de mot comme une unité globale.

- Les techniques de reconnaissance analytiques: techniques basées sur la segmentation d'un mot, une image de document ou une image de mot qui est segmentée en unités plus petites qui peuvent être reconnues indépendamment ou après regroupement partiels.

En outre, nous classifierons les méthodes au sein de chaque catégorie principale selon deux classes:

- Les techniques basées QBE.
- Les techniques basées QBS.

Nous allons maintenant décrire brièvement les méthodes relatives aux différentes classes et leurs sous classes.

## 3.1. Les techniques d'analyse holistique

Les techniques holistiques considèrent chaque image de mot comme une unité. Elles s'appuient principalement sur un processus de segmentation en mots qui doit être réalisée préalablement sur les documents dans lesquels se fait la recherche. En fait, les résultats dépendent de la qualité des mots segmentés dans les images de documents. Nous divisons les techniques holistiques selon la façon dont la requête est formulée : QBE ou QBS.

En fait, Dans la plupart des techniques holistiques, plus précisément dans la sous-catégorie des techniques basées QBE, chaque mot est représenté selon un des trois types de modèles classés en modèles statistiques, modèles pseudo-structurels et modèles structurels. En premier lieu, les modèles statistiques (R. Shekhar, C.V. Jawahar. 2012) (I.Z. Yalniz, R. Manmatha 2012) représentent l'image comme un vecteur de caractéristiques à n dimensions. Ils peuvent être définis à partir des caractéristiques globales et locales. Pour les descripteurs globaux, les caractéristiques scalaires qui sont calculées à partir la totalité de l'image telles que la hauteur, la largeur, le rapport d'aspect, etc… Cependant, les descripteurs locaux comme les SIFT sont calculés à partir des régions locales de l'image ou même à partir des primitives extraites de ces régions. En deuxième lieu, certaines approches holistiques accumulent des informations pseudo-structurelles comme les informations de Loci dans des descripteurs afin de représenter les mots images (D. Fernandez, J. Lladós, A. Fornés. 2011). En troisième lieu, certaines approches décrivent l'image comme une séquence de primitives géométriques et topologiques et génèrent les relations entre elles (P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Liadõs, A. Fornés 2014a).

De plus, la plupart des travaux utilisant une approche holistique basée QBS propose à l'utilisateur d'intervenir après une phase automatique, on les classifie d'approches guidées (A. L. Kesidis, E. Galiotou, B. Gatos and I. Pratikakis. 2011) et d'approches non guidées (D. Aldavert, M. Rusiñol, R. Toledo and J. Lladós. 2013). En fait, plusieurs propositions d'occurrence du mot recherché sont présentées à l'utilisateur qui indique si la proposition convient ou non, c'est du retour de pertinence, et en fonction de ce retour, le système peut améliorer les résultats de spotting.

## 3.2. Les techniques d'analyse analytique

Les techniques analytiques permettent de segmenter l'image de mot ou même une image de document en unités plus petites qui seront reconnues lorsqu'elles sont isolées ou regroupées. En effet, nous dégageons trois catégories de techniques analytiques. Certaines approches basées sur la segmentation exigent que chaque mot soit segmenté en caractères. Ceci pour donner de meilleurs résultats de reconnaissance. D'autres approches utilisent une segmentation explicite des mots en unités plus petites qui sont censées être des portions de caractères qui seront reconnues après. Nous divisons ces approches en deux sous-catégories en fonction de la formulation de la requête comme pour les approches holistiques.

Ainsi, Nous classifions les approches analytiques basées QBE en deux sous-classes qui sont les techniques basées sur la segmentation en caractères (H. Cao and V. Govindaraju. 2007) et les techniques basées sur la non segmentation en caractères (J. Almazán, A. Gordo, A. Fornés, E. Valveny. 2012). Dans la catégorie basée sur la non segmentation en caractères, la plupart des techniques utilisent des fenêtres glissantes afin d'en extraire différentes caractéristiques, par exemple (T. Mondal, N. Ragot, J.Y. Ramel, U. Pal 2015) utilise les techniques de matching DTW. La plupart des travaux qui

s'appuient sur la segmentation des caractères manipulent des documents en chinois ou japonais car la segmentation de l'image de document en caractères est plus facile que dans les autres alphabets.

Par ailleurs, les approches QBS utilisent les techniques de matching DTW et les modèles de HMM et NNs pour résoudre le problème de spotting. Ainsi, nous pouvons identifier deux sous-classes qui sont les techniques basées sur un apprentissage avec un modèle (A. Fischer, A. Keller, V. Frinken, H. Bunke 2012) et les techniques basées sur un apprentissage sans modèle (Y. Liang, M. C. Fairhurst, R. M. Guest. 2012).

Finalement, à partir de cette étude, nous remarquons que les approches analytiques sont très robustes et donnent de meilleurs résultats que les approches holistiques. De plus, fort de l'analyse des travaux précédents, le premier choix que nous avons fait est d'exprimer la requête par la suite de caractères du mot, le plus simple étant une saisie au clavier. En effet, cela permet d'utiliser le système dans toutes les circonstances, même si le mot requête n'est pas présent dans le document. Dans ce cas, on propose une nouvelle technique qui permet de générer une séquence de formes rectangulaires pour chaque requête entrées par l'utilisateur. Cette technique permet, en première approximation, la manipulation des formes rectangulaires au lieu les codes ASCII des requêtes dans le processus de word spotting. Par contre, sans la connaissance des mots du texte, les caractéristiques de l'écriture dans le document ne sont pas connues, parmi ces caractéristiques notons la hauteur du corps des mots. De ce fait, on propose une méthode qui permet de calculer automatiquement la hauteur approximative des caractères dans chaque document manipulé.

Ainsi notre méthode ne repose pas sur un ensemble de caractéristiques fixé a priori et qui indexe les documents dans lesquels se fait la recherche mais sur une famille de caractéristiques qui vont s'adapter simultanément au mot recherché et au document dans lequel se fait la recherche.

Un autre aspect que nous avons voulu éviter est la présence d'une phase d'apprentissage sur une base de documents qui spécialise le système en un système dédié aux documents anciens, aux documents imprimés ou aux documents en anglais par exemple. Il faut pour cela concevoir un système capable d'adaptation et d'auto apprentissage.

# 4. UNE APPROCHE MULTI-ECHELLE POUR LE WORD SPOTTING

Dans notre travail, on se concentre sur les documents manuscrits. Ces documents peuvent représenter une grande variabilité à différents points de vue. Les documents anciens se caractérisent par des variabilités de styles non utilisés de nos jours. Ils se caractérisent aussi par des présentations et des écritures très variées. Les difficultés principales de ce type de documents qu'ils soient anciens ou modernes sont la fragmentation des caractères due souvent à la non homogénéité de l'encre, la variabilité du style de l'écriture et surtout le chevauchement de composantes comme par exemple des composantes appartenant à plusieurs lignes de texte du fait de la présence de hampes et jambages.

A cause de cette variabilité, les systèmes de reconnaissance ne sont pas encore opérationnels. Le word spotting permet sans déchiffrer un texte de retrouver un mot cherché. Dans la méthode que nous proposons, les documents sont traités d'une façon globale, c'est-à-dire sans avoir recours à une phase de segmentation, des lignes ou des mots, phase qui est souvent nécessaire dans les méthodes actuelles de word spotting. La manipulation de l'intégralité du document nécessite souvent un temps de calcul très grand. En tenant compte de toutes ces contraintes, nous proposons une approche comportant 3 grandes phases:

- La représentation de la requête par un ensemble de rectangles.
- Le filtrage global: permettant de réduire très fortement l'espace de recherche en sélectionnant des zones de mots candidats.
- L'affinage: permettant de ne retenir que les mots identiques au mot recherché.

Ces deux dernières grandes phases s'appuient sur la même méthode, un filtrage conçu à 2 échelles différentes.

Comme nous avons opté pour une approche de word spotting dont la requête est constituée d'une suite de caractères saisie au clavier par l'utilisateur, contrairement aux méthodes générant une image synthétisée de la requête, nous allons maintenant décrire l'étape permettant une représentation automatique de chaque requête.

## 4.1. Codage rectangulaire pour la présentation des requêtes

Le principe de la perception humaine nous a aidé à construire une technique automatique de codage des chaînes de caractères. En effet, en regardant un mot par exemple, on peut le caractériser par une forme englobante qui est principalement construite par une suite de formes rectangulaires. Ces rectangles dépendent des caractéristiques de chaque caractère.

Cette technique est alors capable de modéliser grossièrement chaque requête entrée par l'utilisateur par une suite de rectangles dont les tailles dépendent de la taille de chaque élément de la chaîne de caractères. Cette technique de codage n'est pas réversible, c'est-à-dire, qu'à une chaîne de caractères on peut associer sa forme englobante, mais le sens inverse n'est pas possible.

De ce fait, l'idée est de construire un tableau d'index contenant d'une part les symboles et d'autre part des informations. Les deux sorties correspondent aux rectangles de différentes tailles liés à chaque classe et un nombre index qui représente la proportion entre la hauteur du rectangle souhaité et la hauteur d'une lettre minuscule sans hampe et ni jambage se trouvant dans le document manipulé. Pour la création de ce tableau, on va classifier les symboles écrits en différentes classes selon leurs caractéristiques (lettre avec hampe, lettre avec jambage, nombre, ponctuation, etc.).

Ce tableau a 5 classes et un rectangle associé pour chaque classe. Chaque classe contient soit : les lettres majuscules, les lettres minuscules, les nombres, les symboles de ponctuation et l'espace blanc. En outre, la classe contenant les lettres minuscules peut être aussi divisée en 4 sous-classes qui contiennent les lettres avec hampe, les lettres avec jambage, les lettres avec hampe et jambage et le reste de l'alphabet. En conclusion, ce tableau est composé par :

- Classe 1 : lettres minuscules
    - Sous classe 1 : lettres avec hampe : 2
    - Sous classe 2 : lettres avec jambage : -2
    - Sous classe 3 : lettres avec hampe et jambage : 4
    - Sous classe 1 : le reste de l'alphabet : 1
- Classe 2 : lettres majuscules : 2
- Classe 3 : les nombres : 2
- Classe 4 : les symboles : 1
- Classe 5 : l'espace blanc : 0

## 4.2. Le filtrage global

La technique de filtrage global proposée représente la phase majeure de notre méthode. Elle permet d'avoir le moins possible de candidats à l'issue de la première phase de word spotting. A partir d'une requête constituée de caractères tapés au clavier, le filtrage global permet de réduire le nombre de zones possibles qui correspondent plus ou moins à la requête. Le principe de cette partie s'appuie en premier lieu sur un ensemble de points de vue. Sur chaque page de document testée, et en fonction du mot recherché, nous définissons un certain nombre de filtres caractérisant la forme du mot. Pour adapter ces points de vue au document testé, nous tenons compte des caractéristiques de ce dernier. Les caractéristiques du document sont la taille (largeur et hauteur) des caractères. Elle joue un rôle important dans la construction des points de vue et aussi dans l'indexation du tableau de codage. A ce niveau, nous allons maintenant justifier le choix des points de vue et décrire une technique permettant l'estimation automatique de la hauteur moyenne des différents caractères se trouvant dans le document manipulé.

### 4.2.1. Le choix des points de vue

Notre approche est une approche globale qui s'applique à chaque pixel de l'image de document. Donc, le temps de calcul représente une contrainte majeure. Ainsi, les deux étapes de construction et l'application des points de vue ne doivent pas prendre un grand temps d'exécution. Les filtres sont des filtres de Haar calculés instantanément et en complexité constante à partir d'une image intégrale (P. Viola, M. Jones 2001b) En effet, les filtres de Haar sont des filtres rectangulaires asymétriques qui capturent les changements d'intensité dans des zones dont nous choisissons les localisations relatives. Ainsi, on obtient des caractéristiques appelées les caractéristiques pseudo-Haar. Nous combinons ensuite les réponses aux filtres en considérant un accumulateur de votes. Nous obtenons une nouvelle image qui caractérise la présence de mots semblables à la requête dans les documents manipulés (A. Ghorbel, J. M.Ogier, N. Vincent 2015). Ces motifs dépendent des caractéristiques de l'écriture (Lettre minuscule, lettre majuscule, lettre avec hampe ou jambage, etc.).

### 4.2.2. Estimation de la hauteur des minuscules dans des documents manuscrits

La taille des différents caractères minuscules dans les documents manuscrits fait partie de la caractérisation du style de l'écriture.

L'objectif, dans cette étape, n'est de ne pas détecter les lignes de texte mais plutôt d'estimer la hauteur des lignes prédominantes. Ainsi, bien que la forme initiale des filtres dépende de la forme de la requête, la taille des filtres est en relation avec les caractéristiques du document. La hauteur de référence des filtres de Haar est liée à la hauteur du corps de l'écriture. Par conséquent, nous proposons une méthode d'estimation de taille. Cette technique estime automatiquement la taille des lettres minuscules à l'intérieur de chaque image de document.

On applique un ou plusieurs filtres simples de Haar dans le document manipulé en changeant progressivement leurs tailles. Selon la taille de chaque filtre appliqué, on obtient une réponse représentant les lignes de textes dont la hauteur est approximativement la même que celle du filtre. Donc, en changeant les hauteurs des filtres, on peut estimer la hauteur des lettres à partir de la moyenne des hauteurs utilisées. Ainsi, les résultats générés sont alors binarisés et la projection horizontale de chaque image résultat est calculée. A chaque échelle, la valeur médiane des différents pics de la projection horizontale indique l'emplacement des lignes. Finalement, la valeur estimée de la hauteur des caractères mi-

nuscules ne contenant ni hampe ni jambage est calculée par l'intersection entre la courbe des valeurs médianes en fonction des tailles utilisées.

### 4.2.3. Le principe d'accumulation des réponses aux filtres

Après l'application de ces points de vue sur l'intégralité de l'image de document, il s'agit alors de définir les zones d'intérêt. Cette phase est faite globalement au niveau de chaque pixel de l'image du document (A. Ghorbel, J. M.Ogier, N. Vincent 2015). L'étape suivante est de fusionner les points de vue. Elle s'appuie sur un principe de vote. C'est un principe primordial dans notre approche. Pour chaque point de vue appliqué, un vote est exprimé pour l'emplacement spatial d'une zone d'intérêt. La fusion des points de vue correspond à l'accumulation des votes. L'accumulation des hypothèses de la présence du mot nous permet de déduire la présence possible du mot dans les documents testés. L'emplacement spatial de vote associé à un pixel dépend du tableau d'index généré par le processus de codage des requêtes. Ce tableau prédit approximativement la largeur de la requête dans le document manuscrit manipulé. De plus, chaque index dans ce tableau nous aide à décider l'emplacement spatial de chaque point de vue. Egalement, le nombre et la forme généralisée des filtres de Haar qui sont appliqués sont en rapport avec ce tableau. En effet, les indices se trouvant dans ce dernier indiquent la nature de chaque filtre (un filtre pour détecter une lettre avec hampe au début du mot, une lettre avec hampe suivie par une suite de lettres minuscules sans hampes et jambages, etc.). Ainsi, ce tableau d'index joue un rôle important dans l'adaptation et la généralisation des filtres de Haar avec la requête recherchée dans le document manuscrit.

Après l'accumulation des votes, on obtient une image (Iaccum) en niveaux de gris qui met en évidence la présence possible de la requête dans le document manipulé. Ensuite, une étape de binarization est obligatoire pour visualiser les résultats obtenus. Cette étape dépend fortement du seuil de binarization. Ce seuil ne peut pas être fixé à l'avance à cause de la variabilité des styles d'écriture. Il faut que ce seuil soit déterminé automatiquement. On propose une technique qui permet, à partir de l'image Iaccum, de calculer tous les maximums locaux de l'image Iaccum, présentant les réponses générées par l'application des filtres généralisés de Haar à chaque pixel. Ensuite, le seuil est défini comme la valeur moyenne des maximums locaux.

## 4.3. Le filtrage local

Suite aux différentes étapes du filtrage global, nous déduisons la présence possible de beaucoup de candidats qui ressemblent à la requête. Pour réduire le nombre de candidats, nous proposons une deuxième étape qui constitue une phase d'affinage. Cette étape est considérée comme une étude locale dans notre approche qui permet de vérifier si la requête est présente dans le document à cet endroit ou non. Par conséquent, ceci permet de bien affiner les candidats dégagés de filtrage global et permet ainsi de ne retenir que les zones identiques au mot recherché. Finalement, on obtient des candidats qui correspondent le plus à la requête.

Ainsi, nous allons étudier tous les mots sélectionnés et apporter plus de confiance aux mots qui se ressemblent le plus à la requête. La comparaison et la classification sont prises par la représentation des images de mots sélectionnés par des vecteurs de caractéristiques. On a choisi la projection verticale des mots candidats au niveau de gris (T. Rath, S. Kane, A. Lehman, E. Partridge, R. Manmatha 2002). Une classification ascendante hiérarchique (HAC) est ensuite réalisée et qui est basée sur la distance DTW calculée entre les séquences de caractéristiques d'images de mots. Le processus de classification va générer à partir des mots candidats deux classes. En raison du fait que nous ne disposons pas de connaissances *a priori* sur les caractéristiques communes entre la requête et les mots candidats obtenus, nous supposons que la classe contenant plus de candidats est la classe qui contienne plus de

candidats similaires à la requête. Enfin, nous allons garder que cette classe et par conséquent nous allons obtenir que les candidats les plus similaires à la requête. Ceci va augmenter le taux de précision de notre approche.

# 5. LES EXPÉRIMENTATIONS

Dans la partie expérimentale, on a évalué notre approche analytique de word spotting sur deux bases des données publique George Washington (GW) et Bentham. La première base contient des documents du 18ème siècle (T. Rath, R. Manmatha 2007) (V. Lavrenko, T. Rath, R. Manmatha 2004). Cette base présente les caractéristiques suivantes: 20 pages écrites en anglais par 2 scripturs, 656 lignes de textes, 4894 instances de mots, 1471 classes de mots et 82 lettres. En outre, les lettres manuscrites dans cette base ne possèdent pas la même taille vue la variabilité des styles de l'écriture et les lignes sont de différentes épaisseurs. Notre évaluation est comparée avec les deux travaux de word spotting présentés dans (T. Rath, R. Manmatha 2007) et (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007). On a utilisé les mêmes requêtes que celles utilisées dans ces deux travaux et après on a choisi d'une façon aléatoire 100 requêtes de différents mongueurs. Néanmoins, notre approche est basée QBS, donc on n'utilisera pas ces requêtes en tant qu'images mais on les saisira à partir du clavier. La deuxième base de donnée est utilisée dans la compétition « keyword spotting for handwritten documents competition of ICDAR2015 ». Les documents de cette base ont été écrits par le philosophe Jeremy Bentham et ses personnels (1748-1832) (B. Gatos, T. Causer, K. Grint, V. Romero, J. A. Sanchez, A. H. Toselli, E. Vidal 2014). Ils contiennent des mots soulignés et parfois croisés, de bruits et des différents styles d'écritures. Pour le processus de spotting, on a utilisé les 243 requêtes proposées dans la compétition d'ICDAR 2015.

Dans cette partie expérimentale, aucun processus de segmentation de document ni en lignes, ni en mots et ni en caractères n'est utilisé. Ainsi, nous appliquons notre approche globalement sur le document manipulé. De Plus, aucun processus d'apprentissage n'est demandé pour l'appariement entre la requête et les différents candidats.

En outre, nous devons mentionner que nous n'avons supprimé aucun mot ou même ni filtré les ponctuations, comme cela se fait dans quelques travaux de la littérature.

L'évaluation des résultats est faite à travers le taux de rappel, le taux de précision et le taux de F1-mesure.

Premièrement, dans le filtrage globale, et pour les 15 requêtes, on a obtenu une valeur de 96.6% pour le rappel alors que (Y. Leydier,F.LeBourgeois,H.Emptoz. 2007) (T. Rath, R. Manmatha 2007) ont obtenu des taux de rappel 74.2% et 90.72% respectivement. Cependant, le taux de précision n'était pas mentionné dans ces deux articles. Néanmoins, nous générons un taux de précision de 39% et un taux de F1-mesure de 55.81%. On remarque que le taux de précision n'est pas bon pour quelques requêtes. Cela s'explique par le grand nombre des faux positifs qui sont détectés par notre approche. Ces faux positifs correspondent à des candidats qui ont la même forme ou la même taille que la requête passée en entrée. En outre, la qualité et l'épaisseur de trait de chaque caractère entrent en jeu et affectent les résultats générés par les filtres généralisés de Haar. Après le filtrage locale, le taux de rappel est resté le même tandis que la précision et le F1 score sont augmentés et ils sont de 61% et 74.77% respectivement. En plus, le taux de rappel, de précision et de F1 score obtenu pour les 100 requêtes après les deux phases de filtrages sont de 95.6%, 78.2 %et 86.02 % respectivement. Cependant, pour la recherche des 243 requêtes dans la base de Bentham, on obtient des valeurs de rappel, précision et de f1 score de 83.64%, 58.32% et 68.72% respectivement.

# 6. EXTRACTION DES TEXTES DANS LES BANDES DESSINEES

## 6.1. Introduction

Les bandes dessinées constituent une part non négligeable du patrimoine culturel dans de nombreux pays. Le divertissement et les contes peuvent être considérés comme un art en soi. Ils représentent une œuvre artistique pleine d'imagination et de créativité. Un des dispositifs typiques de la narration est la bande dessinée. Ceux-ci ont été présents avec différentes représentations dans la culture humaine depuis de nombreuses années. De nos jours, en prenant en considération les avancées technologiques, les bandes dessinées sont proposées pour être téléchargées sur les appareils numériques tels que les appareils mobiles. Ainsi, les bandes dessinées ont été largement impliquées en particulier dans nos moments de loisir et de vie. Pourtant, ceci introduit la nécessité de processus automatiques pour traiter les contenus des BD.

Par conséquent, cette évolution des moyens de diffusion a encouragé la communauté des chercheurs dans le domaine des documents graphiques à analyser la grande quantité d'albums de bandes dessinées. Différentes applications ont été proposées Durant plusieurs années pour les documents graphiques tels que l'indexation, la recherche d'éléments spécifiques, et l'analyse de contenu (par exemple : la localisation de texte, la détection des bulles, et la localisation et l'interprétation des cases).

Dans la bande dessinée, le texte présente certaines caractéristiques. Dans une vision grossière d'une page graphique, il présente quelques pièces bien contrastées horizontales, quelles que soient les couleurs du texte ou de l'arrière-plan sont. En regardant la page entière, le texte apporte des contours horizontaux.

Cependant, peu d'études ont été menées à ce jour. Nous allons mentionner les études de l'état de l'art qui ont analysé les bandes dessinées.

## 6.2. Etat de l'art

Les comics sont connus et classés en trois catégories. Cette classification est basée sur les particularités culturelles de chaque pays et de sa langue parlée. Ils sont considérés comme des documents graphiques complexes. Leurs fonds sont de nature graphique. Les éléments textuels peuvent être situés partout dans la page et orientés avec des polices différentes, le style, l'alignement, la taille et les couleurs. Chaque page contient une séquence de trames ou de bandes qui sont séparées par des tubes ou des gouttières blancs ou mono-couleur. En effet, nous pouvons également trouver d'autres contenus tels que des bulles et des dessins. Le texte peut être soit contenu dans chaque cadre soit être relatif à deux cadres ou plus.

Plus particulièrement, le texte peut être écrit soit à l'intérieur des bulles, soit à l'extérieur. En effet, la plupart des techniques de l'état d'art qui traitent la détection et la localisation de texte fondent leurs travaux sur des hypothèses qui sont 'texte fait partie de bulles de la parole' ou 'texte est écrit en noir dans une bulle de fond blanc'. Ainsi, la détection de trame et des bulles sont deux étapes de détection qui doivent être réalisées avant l'étape d'extraction de texte.

Les méthodes de séparation texte / zones graphiques pour les documents de bandes dessinées peuvent être classées en deux catégories différentes dans la littérature: les approches top-down et bottom-up. Pour la première catégorie, la plupart des approches top-down ou les approches descendantes commencent par la détection des trames ou les bulles et après elles utilisent des techniques de détection et de localisation des lignes de textes dans ces bulles. La limitation de ces approches est que seulement les bulles fermées sont détectées. D'autre part, les approches bottom-up ou les approches ascendantes sont basées sur les techniques de composantes connexes et des techniques de segmentation. Elles ne nécessitent pas une segmentation de trames ou des bulles de paroles *a priori*. Ces travaux possèdent des limites : soit ils sont dépendant de l'orientation et de la résolution, soit ils assument que la page contient des textes et la couleur du texte est similaire que le couleur de l'arrière-plan de la page.

En outre, nous pouvons aussi penser à la détection de texte, plans graphiques d'architectures ou autres. La complexité de l'image dans les bandes dessinées est encore plus élevée, mais le nombre de mots de texte est généralement plus grand. Du coup, certaines méthodes spécifiques sont nécessaires.

Dans le cadre de notre travail, nous proposons une méthode automatique d'extraction de texte pour les bandes dessinées numérisées qui ne repose pas sur l'une des hypothèses précédentes.

## 6.3. L'approche proposée

Dans cette section, nous allons décrire le processus de détection de texte et sa localisation dans les bandes dessinées.

### 6.3.1. La détection du texte

Comme mentionné dans l'introduction de cette partie, la plus grande partie des régions de texte introduit des parties bien contrastées horizontales, quelles que soient les couleurs du texte ou de l'arrière-plan. Ainsi, les différentes parties de texte contribuent à des contours horizontaux.

Donc, pour détecter les zones de texte à partir de la bande dessinée, l'idée est de développer un extracteur de contour approprié. Vu que nous modélisons une version grossière du document, seule la direction et le contraste sont importants. Ainsi, nous avons choisi d'appliquer les caractéristiques pseudo-Haar qui sont facilement calculées à partir de l'image intégrale.

Le principe fondamental de cette technique est basée sur l'application à l'échelle globale d'un certain nombre de points de vue qui sont modélisés par des filtres de Haar appliqués sur les images de bandes dessinées. En fait, ils correspondent à une transformation globale de l'image BD qui permet de mettre en évidence des formes différentes. Ces points de vue sont traités de façons indépendantes et leurs conclusions sont rassemblées pour générer les résultats finaux.

En effet, les filtres appliqués doivent être construits en tenant compte d'une caractéristique majeure qui est la taille de l'écriture tout au long des bandes dessinées. Mais, cette taille n'a pas à être estimée de façon précise, seuls les contours doivent être détectés. Puis, après avoir défini les filtres qui sont caractérisés par un noyau (k), nous avons appliqué chaque noyau sur l'image traitée. Nous obtenons une image transformée Ik. Cette image transformée souligne plusieurs des régions qui représentent les régions de texte dans l'image. Ensuite, nous accumulons les informations de chaque filtre appliqué afin d'obtenir des zones d'intérêt. Les résultats finaux sont générés à partir d'une binarization de l'image accumulée.

Dans ce travail, nous définissons certains filtres asymétriques rectangulaires de Haar et qui permettent la détection des zones de texte horizontales alignées, soit la partie supérieure de texte ou la partie inférieure du texte. La hauteur de ces filtres est définie *a priori* à 20 pixels parce que les textes sont lisibles et qui sont écrites en petite taille dans la plupart des bandes dessinées.

### 6.3.2. La localisation du texte dans les bandes dessinées

Les résultats de la phase de détection génèrent de nombreuses zones de candidats, certains sont des faux positifs qui correspondent plus ou moins au texte car ils contiennent des graphiques horizontaux.

Ainsi, une étape de filtrage basée sur les composantes connexes est appliquée. Tout d'abord, les boîtes englobantes associées aux candidats de texte sont extraites. Deux critères permettent de sélectionner les boîtes englobantes contenant les zones de texte. Les boîtes englobantes ne doivent pas être trop grandes et elles devraient être associées à des lignes ou des mots, elles doivent avoir un rapport d'aspect élevé (largeur / hauteur). Ensuite, on ne retient que les boîtes englobantes avec une surface inférieure à la médiane des surfaces des boîtes englobantes. Après avoir décrit l'approche proposée, nous allons maintenant présenter les résultats expérimentaux.

## 6.4. Les expérimentations

Notre approche de détection et localisation de texte dans les bandes dessinées a été évaluée sur la base de données BDtheque. Cette base contient 100 pages de comics et elle est caractérisée par la diversité des formats et des styles. En outre, les pages de bandes dessinées diffèrent selon les techniques de conception et d'impression et elles ont été publiées dans le 20 et 21ème siècle. Nous évaluons notre approche en termes de boîtes englobantes des objets.

D'après les résultats obtenus, nous concluons que notre approche surmonte certaines contraintes introduites par la littérature tels que: ‘le texte est écrit en noir dans une bulle blanche’; ‘Couleur de fond du texte devrait être similaire à l’arrière-plan de la page’; ‘Texte est contenu dans une bulle’.

Pourtant, l’approche ne se fonde pas sur des hypothèses concernant la localisation, la couleur et le style d’écriture du texte dans les comics. En effet, cette approche n'a pas besoin de techniques de segmentation des frames ou de détection des bulles pour localiser les textes et graphiques. Un autre point fort est que cette méthode permet de détecter le texte qui est soit à l'intérieur soit à l'extérieur des bulles et de générer des résultats précis. Cependant, les résultats qualitatifs montrent que notre approche échoue quand il y a des régions de texte de grande courbure ou dans le cas où le texte est plus clair que le fond.

En outre, les résultats quantitatifs sont obtenus en évaluant notre approche sur 4667 lignes de texte de l'ensemble de données BDtheque au niveau des boîtes englobantes des objets. Toutes les pages de comics sont traitées dans le processus d'évaluation, mais en excluant seulement 6 pages de manga japonais. Les pages de manga japonais sont traités avec d'autres rectangles asymétriques de Haar qui ne sont pas fondées sur l'horizontalité, mais sur la verticalité des caractéristiques de l'écriture japonaise. Ainsi, l'évaluation se fait sur le sous-ensemble de textes en anglais en en français avec l'alphabet latin.

Cependant, à partir des résultats qualitatifs, nous remarquons qu'il y a des faux positifs et des faux négatifs qui vont influencer les résultats quantitatifs. Donc, afin d'éliminer autant que possible les faux positifs, nous avons effectué une technique de filtrage de post-traitement après l'étape d'extraction de texte dans les images obtenues. L'étape de filtrage est basée sur les composants connexes. Dans cette

étape, nous avons affaire à des images résultantes de l'étape d'extraction du texte et du graphique. Ainsi, les boîtes englobantes ayant une valeur d'aspect ratio faible en ce qui concerne la taille des textes détectés sont éliminées.

Nous avons évalué l'extraction de texte en utilisant différents seuils N qui correspondent au pourcentage minimum de chevauchement entre les régions de texte à détecter et les régions détecté. L'évaluation est faite au niveau de la ligne de texte. Les lignes de texte qui sont transcrites fidèlement par rapport la vérité terrain sont considérées comme des transcriptions correctes. Les meilleurs résultats sont obtenus lorsque N est égal à 30%.

La valeur optimale de N qui offre une meilleure précision et rappel (R = 93%, P = 32%) est de 30%. Cela signifie que certaines régions de vérité terrain qui sont détectés comme correctes ont été détectées seulement à 30% de leur superficie. Cela est dû à la faible densité du texte dans une zone de sélection de texte.

Nous remarquons que notre approche génère un taux de rappel élevé. En effet, ceci est le meilleur taux de rappel à l'état de l'art. Enfin, le F1-score (47.62%) obtenu par notre approche est presque similaire que les F1-score des approches proposées dans l'état de l'art. Cela confirme la justesse de notre approche proposée pour le texte et la séparation graphique dans les bandes dessinée.

# 7. CONCLUSION

Dans cette thèse, nous avons proposé une approche analytique qui combine les échelles pour le word spotting dans les documents manuscrits. Le modèle proposé fonctionne selon deux niveaux différents. Un module de filtrage global permettant de définir plusieurs zones candidates de la requête dans le document testé. Ensuite, l'échelle de l'observation est modifiée à un niveau inférieur afin d'affiner les résultats et sélectionner uniquement ceux qui sont vraiment pertinents. Vu que notre approche manipule des requêtes au format ASCII, on a proposé une technique automatique de codage permettant de représenter chaque requête par des formes rectangulaires et des indexes. Cette approche de word spotting est basée sur des familles généralisées de filtres de Haar qui s'adaptent à chaque requête pour procéder au processus de spotting et aussi sur un principe de vote qui permet de choisir l'emplacement spatial où les réponses générées par les filtres sont accumulées. De plus, une technique automatique d'estimation de la hauteur moyenne des caractères écrits dans chaque document manuscrit est décrite assurant l'adaptation aux caractéristiques des documents. Finalement, les résultats menés sur la base de donnée GW montrent que le taux moyen de rappel est très bon par rapport l'état de l'art. Nous avons en plus proposé une autre approche pour l'extraction de texte du graphique dans les bandes dessinées. Cette approche se base essentiellement sur les caractéristiques pseudo-Haar qui sont générées par l'application des filtres généralisés de Haar sur l'image de bande dessinée. Cette approche ne nécessite aucun processus d'extraction ni des bulles de conversation ni d'autre composant. En effet, elle s'applique d'une façon globale et elle génère de bons résultats.

Nous avons aussi présenté une méthode de détection et localisation automatique de texte dans les bandes dessinées. Cette méthode est également basée sur l'application de filtres de Haar pour détecter les zones de texte et sur un algorithme d'étiquetage des composantes connexes pour extraire les zones de texte. L'originalité de ce travail est qu'il ne repose pas sur des hypothèses concernant la localisation et la couleur et le style d'écriture du texte dans les comics. En effet, cette étape se fait à l'échelle globale sur l'image RVB sans appliquer les techniques de détection des frames ou les ballons de parole. Les expériences montrent que cette méthode permet de détecter le texte à l'intérieur ou à l'extérieur les bulles et de générer des résultats précis presque similaires à l'état-of-the-art avec les meilleurs taux de rappel.

# VIII. Author publications

[1] Ghorbel, A., Ogier, J.. & Vincent, N (2016)
“Adaptation des caractéristiques pseudo-Haar pour le word spotting dans les documents manuscrits”,
In CORIA 2016 - Conférence en Recherche d'Informations et Applications- 13th French Information Retrieval Conference. CIFED 2016 Colloque International Francophone sur l'Ecrit et le Document, Toulouse, France, March 9-11, 2016, Toulouse, France, March 9-11, 2016., pages 497-512

[2] Ghorbel, A., Ogier, J. & Vincent, N (2015)
“Haar-like-features for query-by-string word spotting”,
In 17th Biennial Conference of the International Graphonomics Society

[3] Ghorbel, A., Ogier, J. & Vincent, N (2015)
Text extraction from comic books,
In GREC 2015 Eleventh IAPR International Workshop on Graphics Recognition

[4] Ghorbel, A., Ogier, J.. & Vincent, N (2015)
“A segmentation free Word Spotting for handwritten documents”,
In 13th International Conference on Document Analysis and Recognition, ICDAR 2015, Tunis, Tunisia, August 23-26, 2015, pages 346-350

[5] Ghorbel, A., Ogier, J.. & Vincent, N (2015)
“Word Spotting techniques for documents images: A comprehensive survey”,
In International Journal on Document Analysis and Recognition (IJDAR) (Under Review)

*"We can only see a short distance ahead, but we can see plenty there that needs to be done"* **_Alan Turing**

# IX. References

A. Andreev and N, Kirov. 2009. “Word Image Matching Based on Hausdorff Distances.” *10th Int. Conf. Doc. Anal. Recognit.*

A. Balasubramanian , Balasubramanian Million Meshesha , C. V. Jawahar. 2006. “Retrieval from Document Image Collections.” *Proc DAS*, 1–12.

A. Bhardwaj, D. Jose and V. Govindaraju. 2008. “Script Independent Word Spotting in Multilingual Documents.” *Proc. 2nd Int. Workshop Cross Lingual Inf. Access*, 48–54.

A. de Bouard. 1911. “Latin Handwriting.” *Present in Chisholm, Hugh, Ed. (1911), Encyclopædia Britannica Eleventh Edition, Cambridge University Press*.

A. Fischer, A. Keller, V. Frinken, and H. Bunke. 2010. “HMM Based Word Spotting in Handwritten Documents Using Subword Models.” *Proc. of the Int. Conf. on Pattern Recognition:*, 3416–3419.

A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012. “Lexicon-Free Handwritten Word Spotting Using Character HMMs.” *Pattern Recognition Letters Volume 33 Issue 7:*, 934–942.

A. Fischer, A. Keller, V. Frinken, H. Bunke. 2012. “Lexicon-Free Handwritten Word Spotting Using Character HMMs.” *Pattern Recognition Letters Volume 33 Issue 7*, 934–42.

A. Fournier. 1996. “Wavelets and Their Applications in Computer Graphics.” *SIGGRAPH’95 Course Notes, University of British Columbia*.

A. G. Fink, L. Rothacker, R. Grzeszick. 2014. “Grouping Historical Postcards Using Query-by-Example Word Spotting.” *14th International Conference on Frontiers in Handwriting Recognition*.

A. Ghorbel, J. M.Ogier, N. Vincent. 2015. “A Segmentation Free Word Spotting for Handwritten Documents.” *ICDAR 2015*, August.

A. Haar. 1910. “Zur Theorie Der Orthogonalen Funktionensysteme.” *Mathematische Annalen* 69 (3): 331–371.

A. H. Toselli, E. Vidal. 2013. “Fast HMM-Filler Approach for Key Word Spotting in Handwritten Documents.” *In Document Analysis and Recognition (ICDAR) 12th International Conference on*, 501–5.

———. 2014. “Word-Graph Based Handwritting Key-Word SPotting: Impact of Word-Graph Size on Performance.” *In Document Analysis Systems (DAS) 11th IAPR International Workshop on*, 176–80.

A. H. Toselli, E. Vidal, V. Romero, V. Frinken. 2013a. “Word-Graph Based Keyword Spotting and Indexing of Handwritten Document Images.” *Technical Report, Universitat Polite ´Cnica de Vale ´Ncia*.

———. 2013b. “Word-Graph Based Keyword Spotting in Handwritten Document Images.” *Universidad Politécnica de Valencia. Tech. Rep*.

A. K. Ngo Ho, J. C.Burie, J. M.Ogier. 2011. “Comics Page Structure Analysis Based on Automatic Panel Extraction.” *In: Grec, Nineth IAPR International Workshop on Graphics Recognition*.

A. Khammari, E. Lacroix, F. Nashashibi, C. Laurgeau. 2005. “Vehicle Detection Combining Gradient Analysis and AdaBoost Classification.” *IEEE Conferences on Intelligent Transportation Systems, Vienna*, 66–71.

A. Kołcz, J. Alspector, M. F. Augusteijn, R. Carlson, and G. V. Popescu. 2000. “A Line-Oriented Approach to Word Spotting in Handwritten Documents.” *Pattern Analysis and Applications*, 153–68.

A. L. C. Barczak. 2005. “Toward an Efficient Implementation of a Rotation Invariant Detector Using Haar-Like Features.” *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*.

A. L. C. Barczak , F. Dadgostar , C. H. Messom. 2005. “Real-Time Hand Tracking Based on Noninvarient Features.” *IEEE Instrumentation and Measurement Technology Conference*, 2192–97.

A. L. Kesidis, E. Galiotou, B. Gatos and I. Pratikakis. 2011. “A Word Spotting Framework for Historical Machine-Printed Documents.” *Int. J. Doc. Anal. Recogn.*

A. Mishra, K. Alahari, C. V. Jawahar. 2012. “Scene Text Recognition Using Higher Order Language Priors.” *In BMVC*.

A. P. Giotis, G. Sfikas, C. Nikou, B. Gatos. 2015. “Shape-Based Word Spotting in Handwritten Document Images.” *13th IAPR International Conference on Document Analysis and Recognition (ICDAR 2015)*, August.

A. Shrivastava, T. Malisiewicz, A. Gupta, A.A. Efros. 2011. “Datadriven Visual Similarity for Cross Domain Image Matching.” *ACM Trans. Graph. TOG Proc. ACM SIGGRAPH ASIA*.

“Annales de L’insée.” 1980, no. 40.

B. Bley. n.d. “Feldpostkarten Im 1. Weltkrieg (Feldspot Postcards of World War I).” *Private Collection, Dortmund, Germany*.

B. Gatos and I. Pratikakis. 2009. “Segmentation-Free Word Spotting in Historical Printed Documents.” *ICDAR*.

B. Gatos., N. Papamarkos, C, Chamzas. 1997. “A Binary Tree Based OCR Technique for Machine Printed Characters.” *Eng Appl Artif Intell 104*, 403–12.

B. Gatos, T. Causer, K. Grint, V. Romero, J. A. Sanchez, A. H. Toselli, E. Vidal. 2014. “Ground-Truth Production in the Transcriptorium Project.” *In 11th IAPR International Workshop on Document Analysis System (DAS)*.

B. Gatos, T. Konidaris, K. Ntzios, I. Pratikakis, S. J. Perantonis. 2005. “A Segmentation-Free Approach for Keyword Search in Historical Typewritten Documents.” *Proceeding Eighth Int. Conf. Doc. Anal. Recognit. ICDAR,*.

C. Guerin, C. Rigaud, A. Mercier, F. Ammar-Boudjelal, K. Bertet, A. Bouju, J. C. Burie, G. Louis, J. M.Ogier, A. Revel. 2013. “eBDtheque: A Representative Database of Comics.” *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR)*.

C. H. Messom, A. L. C. Barczak. 2009. “Stream Processing for Fast and Efficient Rotated Haar-like Features Using Rotated Integral Images.” *IJISTA* 7 (1): 40–57.

C. Ponsard, V. Fries. 2008. “An Accessible Viewer for Digital Comic Books.” *In: ICCHP, LNCS 5105*.

C. Rigaud, C. Guérin, D. Karatzas, J. C. Burie, and J. M. Ogier. 2015. “Knowledge-Driven Understanding of Images in Comic Books.” *International Journal on Document Analysis and Recognition (IJDAR)*.

C. Rigaud, D. Karatzas, J. Weijer, J. C. Burie, and J. M. Ogier. 2013. “Automatic Text Localisation in Scanned Comic Books.” *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*, 814--819.

C. Rigaud, N. Tsopze, J. C. Burie, J. M.Ogier. 2011. “Robust Frame and Text Extraction from Comic Books.” *In: Grec, Nineth IAPR International Workshop on Graphics Recognition*.

C. Y. Su, R. I. Chang, and J. C. Liu. 2011. “Recognizing Text Elements for Svg Comic Compression and Its Novel Application.” *In Proceedings of the 2011 International Conference on Document Analysis and Recognition , ICDAR ’11*, 329–1333.

Cyb. 2009. “Bubblegôm.” *Studio Cyborga*.

D. Aldavert, M. Rusiñol, R. Toledo and J. Lladós. 2013. “Integrating Visual and Textual Cues for Query-by-String Word Spotting.” *In Proceedings of the Twelfth International Conference on Document Analysis and Recognition, ICDAR.*, 511–15.

D.A. Lisin, M.A. Mattar, M.B. Blaschko, M.C. Benfield,E.G. Learned-miller. 2005. “Combining Local and Global Image Features for Object Class Recognition.” *Proc. IEEE CVPR Workshop Learn. Comput. Vis. Pattern Recognit.*, 47–55.

D. Fernandez, J. Lladós, A. Fornés. 2011. “Handwritten Word Spotting in Old Manuscript Images Using a Pseudo-Structural Descriptor Organized in a Hash Structure.” *Iberian Conference on Pattern Recognition and Image Analysis:*, 628–35.

D. Fernãndez, P. Riba, A. Fornés, J. Lladõs. 2014. “On the Influence of Key Point Encoding for Handwritten Word Spotting.” *In 14th International Conference on Frontiers in Handwritting Recognition*.

D. G. Long. 1981. “The Manuscripts of Jeremy Bentham: A Chronological Index to the Collection in the Library of University College, London: Based on the Catalogue by A. Taylor Milne.” *The College*.

D. Sankoff, J. Kruskal. 1999. “Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparision.” *CSLI Publications*.

D. Yankowitz, A. M. Bruckstein. 1989. “New Method for Image Segmentation.” *Computer Vision, Graphics and Image Processing*, *46/1*, 82–95.

E. J. Keogh, M. J. Pazzani. 2001. “Derivative Dynamic Time Warping.” *In First SIAM International Conference on Data Mining*.

F. Crow. 1984. “Summed-Area Tables for Texture Mapping.” *SIGGRAPH* 84: 207–12.

F. H. Harmuth. 1978. “Sequency Theory.” *Foundations and Applica Tions. Academic Press, New York.*

F. Moreno-Noguer. 2011. “Deformation and Illumination Invariant Feature Point Descriptors.” *In Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, 1593–1600.

F. R. Chen, D.S Bloomberg, L.D Wilcox.,. 1995. “Spotting Phrases in Lines of Imaged Text.” *Proc. SPIE Doc. Recognit. II Int. Soc. Opt. Eng. SPIE°*.

G. A. Ramirez, O. Fuentes. 2008. “Multi-Pose Face Detection with Asymmetric Haar Features.” *Application of Computer Vision, 2008*.

G. Khaissidi, Y. Elfakir, M. Mrabti, M. El Yacoubi, D. Chenouni, Z. Lakhliai. 2016. “Segmentation-Free Word Spotting for Handwritten Arabic Documents.” *Journal Article Published 2016 in International Journal of Interactive Multimedia and Artificial Intelligence Volume 4 Issue 1*.

H. A. Glucksman. 1967. “Classification of Mixed-Font Alphabetics by Characteristic Loci.” *Dig. First Annu. IEEE Comput. Conf.*, 138–41.

H. Cao and V. Govindaraju. 2007. “Template-Free Word Spotting in LowQuality Manuscripts.” *In 6th Int’l Conf. on Advances in Pattern Recognition.*

H. Chatbri, P. Kwan, K. Kameyama. 2014. “An Application-Independent and Segmentation-Free Approach for Spotting Queries in Document Images.” *22nd International Conference on Pattern Recognition*.

I.Z. Yalniz, R. Manmatha. 2012. “An Efficient Framework for Searching Text in Noisy Document Images.” *Document Analysis Systems, IEEE*, 48–52.

J. Almazán, A. Gordo, A. Fornés, E. Valveny. 2012. “Efficient Exemplar Word Spotting.” *BMVC*.
J. Almazán, A. Gordo, A. Fornés, E. Valveny. 2013. “Handwritten Word Spotting with Corrected Attributes.” *In Computer Vision (ICCV), 2013 IEEE International Conference on*, December, 1017–42.
———. 2014a. “Segmentation-Free Word Spotting with Exemplar SVMs.” *Pattern Recognition Journal, 47 (12)*, 3967–78.
———. 2014b. “Word Spotting and Recognition with Embedded Attributes.” *IEEE Transactions on Pattern Analysis and Machione Intelligence*.
J. Almazãn, D. Fernãndez, A. Fornés, J. Lladõs, E. Valveny. 2012. “A Coarse-to-Fine Approach for Handwritten Word Spotting in Large Scale Historical Documents Collection.” *In Proc. of ICFHR*, 453–58.
J.A. Rodriguez-Serrano, F. Perronnin. 2009. “Handwritten Word-Image Retrieval with Synthesized Typed Queries.” *10th Int. Conf. Doc. Anal. Recognit.*
J.A. Rodríguez-Serrano, F. Perronnin. 2009. “Handwritten Word-Spotting Using Hidden Markov Models and Universal Vocabularies.” *Pattern Recognit.*, 2106–16.
J. Barreto, P. Menezes, J. Dias. 2004. “Human-Robot Interaction Based on Haar-like Features and Eigenfaces.” *International Conference on Robotics and Automation*.
J. J. Marcos. 2011. “Fonts for Latin Palaeography.” *Encyclopædia Britannica Eleventh Edition.*
J. Li, Z.G. Fan, Y. Wu ,N. Le. 2009. “Document Image Retrieval with Local Feature Sequences.” *10th International Conference on Document Analysis and Recognition.*
J. Lladós, M. Rusiñol, A. Fornés, D. Fernández and A. Dutta. 2012. “On the Influence of Word Representation for Hadwritten Word Spotting in Historical Documents.” *International Journal of Pattern Recognition and Artificial Intelligence*.
J. Puigcerver, A. h Toselli, E. Vidal. 2014. “Word-Graph-Based Handwriting Keyword Spotting of out-of-Vocabulary Queries.” *In: 22nd International Conference on Pattern Recognition (ICPR)*, 2035–40.
J. Puigcerver, A. H. Toselli, E. Vidal. 2015. “ICDAR 2015 Competition on Keyword Spotting for Handwritten Documents.” *13th International Conference on Document Analysis and Recognition ICDAR 2015*.
J. Puigcerver, A. Toselli, E. Vidal. 2016. “Querying out-of-Vocabulary Words in Lexicon-Based Keyword Spotting.” *Journal Article Published 3 Feb 2016 in Neural Computing and Applications*.
J. Rodríguez-Serrano and F. Perronnin. 2008. “Local Gradient Histogram Features for Word Spotting in Unconstrained Handwritten Documents.” *ICFHR*.
———. 2012. “A Model-Based Sequence Similarity with Application to Handwritten Word-Spotting.” *IEEE TPAMI*.
J.L Rothfeder, S. Feng, T.M. Rath. 2003. “Using Corner Feature Correspondences to Rank Word Images by Similarity.” *Conference on Computer Vision and Pattern Recognition Workshop*.
K. Arai, H. Tolle. 2010a. “Method for Automatic E-Comic Scene Frame Extraction for Reading Comic on Mobile Devices.” *In: Seventh International Conference on Information Technology: New Generations*.
———. 2010b. “Automatic E-Comic Content Adaptation.” *International Journal of Ubiquitous Computing IJUVC*, May.
———. 2011. “Method for Real Time Text Extraction of Digital Manga Comic.” *International Journal of Image Processing (IJIP)*.
K. Khurshid, C. Faure, N. Vincent. 2009. “A Novel Approach for Word Spotting Using Merge-Split Edit Distance.” *Computer Analysis of Images and Patterns, 13th International Conference, CAIP 5702*, 213–20.
K. Terasawa, and Y. Tanaka. 2007. “Locality Sensitive Pseudocode for Document Images.” *IEEE Proc. Int. Conf. Doc. Anal. Recognit. ICDAR’07*, 73–77.
K. Terasawa and Y. Tanaka. 2009. “Slit Style HOG Features for Document Image Word Spotting.” *10th Int’l Conf Doc. Anal. Recognit.*, 116–20.
K. Terasawa,H. Imura, Y. Tanaka. 2009. “Automatic Evaluation Framework for Word Spotting.” *10th Int. Conf. Doc. Anal. Recognit. ICDAR 09*, 276 – 280.
K. Terasawa,T.Nagasaki,T.Kawashima. 2005. “Eigenspace Method for Text Retrieval in Historical Document Images.” *8th International Conference on Document Analysis and Recognition(ICDAR).*
K. Wang, B. Babenko, S. Belongie. 2011. “End-to-End Scene Text Recognition.” *In ICCV*.
K. Zagoris, I. Pratikakis, B. Gatos. n.d. “Segmentation-Based Historical Handwritten Word Spotting Using Document-Specific Local Features.” *14th International Conference on Frontiers in Handwriting Recognition*, no. 2014.
K. Zagoris, N. Papamarkos, C. Chamzas. 2006. “Web Document Image Retrieval System Based on Word Spotting.” *IEEE Int. Conf. Image Process.*
L. Daubechies, W. Sweldens. 1998. “Factoring Wavelet Transforms into Lifting Steps.” *J. Fourier Anal. Appl., 4(3)*, 247–69.

L. Rothacker, G. A. Fink, P. Banerjee, U. Bhattacharaya, B. B. Chaudhuri. 2013. “Bag-of-Features HMMs for Segmentation-Free Bangla Word Spotting.” *In Proceedings of the 4th International Workshop on Multilingual OCR, ACM*.

L. Rothacker, L. Vajda, G. A. Fink. 2012. “Bag-of-Features Representations for Offline Handwritting Recognition Applied to Arabic Script.” *In ICFHR*, September, 149–54.

L. Rothacker, M. Rusinol, G. A. Fink. 2013. “Bag-of-Features HMMs for Segmentation-Free Word Spotting in Handwritten Documents.” *In Document Analysis and Recognition (ICDAR), 12th INTERNATIONAL cONFERENCE ON*, August, 1305–9.

M. H. A. Newman. 1955. “Alan Mathison Turing. 1912–1954.” *Biographical Memoirs of Fellows of the Royal Society* 1: 253–226.

M. Jones, P. Viola. 2003. “Fast Multi-View Face Detection.” *Mitsubishi Electric Research Lab TR-20003-96*.

M. Jones, P. Viola, D. Snow. 2003. “Detecting Pedestrians Using Patterns of Motion and Appearance.” *Technical Report, Mitsubishi Electric Research Laboratories, TR200390*.

M. Kassis, J. El-Sana. 2014. “Word Spotting Using Radial Descriptor.” *14th International Conference on Frontiers in Handwriting Recognition*.

M. Keyvanpour, R. Tavoli, S. Mozaffari. 2013. “Document Image Retrieval Based on Keyword Spotting Using Relevance Feedback.” *International Journal of Engineering-Transaction A: Basics 27(1)*, 7–14.

M. Kolsch, M. Turk. 2004. “Analysis of Rotational Robustness of Hand Detection with a Viola-Jones Detector.” *In ICPR04*, 107–10.

M. Makridis,N.Nikolaou,B.Gatos. 2007. “An Efficient Word Segmentation Technique for Historical and Degraded Machine-Printed Documents.” *International Conference on Document Analysis and Recognition (ICDAR), Curitiba, Brazil*, 178–182.

M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, T. Poggio. 1997. “Pedestrian Detection Using Wavelet Template.” *In Computer Vision and Pattern Recognition*, 193–99.

M. Rusinõl , J. Llados. 2008. “Word and Symbol Spotting Using Spatial Organization of Local Descriptors.” *In The Eighth IAPR Workshop on Document Analysis Systems.*

M. Rusinõl , J. Llados. 2009. “A Performance Evaluation Protocol for Symbol Spotting Systems in Terms of Recognition and Location Indices.” *Int. J. Document Analysis and Recognition* 12 (2): 83–96.

———. 2014. “Boosting the Handwritten Word Spotting Experience by Including the User in the Loop.” *Pattern Recognition Journal, 47 (3)*.

M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós. 2011. “Browsing Heterogeneous Document Collections by a Segmentation-Free Word Spotting Method.” *ICDAR*.

M. Rusiñol, D. Aldavert, R. Toledo and J. Lladós. 2015. “Efficient Segmentation-Free Keyword Spotting in Historical Document Collections.” *Pattern Recognition 48(2)*, 545–55.

M. Yamada, R. Budiarto, M. Endo, S. Miyazaki. 2004. “Comic Image Decomposition for Reading Comics on Cellular Phones.” *IEICE Transactions*.

N. Doulgeri,E. Kavallieratou. 2009. “Retrieval of Historical Documents by Word Spotting.” *16th Doc. Recognit. Retr. Conf. DRR-09 USA*.

N. J. Pyun, H. Sayah, N. Vincent. 2014. “Adaptative Haar-like Features for Head Pose Estimation.” *11th International Conference ICIAR 2014*, 94–101.

O. S. Jahromi, B. A. Francis, R. H. Kwong. 2003. “Algebraic Theory of Optimal Filterbanks.” *EEE Transactions on Signal Processing, 51*, 442–57.

P. Bilane , S. Bres , K. Challita , H. Emptoz. 2009. “Indexation of Syriac Manuscripts Using Directional Features.” *Int. Conf. Image Process. ICIP*, 1841–1844.

P. Gray. 1999. “Alan Turing – Time 100 People of the Century.” *Time Magazine*.

P. Jorgensen. 2003. “Matrix Factorizations, Algorithms, Wavelets.” *Notices of the American Mathematical Society, 50(8)*, 880–94.

P. Keaton, H. Greenspan, R. Goodman. 1997. “Keyword Spotting for Cursive Document Retrieval.” *In: Workshop on Document Image Analysis (DIA 1997)*, 74–82.

P. Negri, X. Clady, S. M. Hanif, L. Prevost. 2008. “A Cascade of Boosted Generative and Discriminative Classifiers for Vehicle Detection.” *EURASIP Journal on Advances in Signal Processing*.

P. porwik, A. Lisowska. 2004a. “The Haar-Wavelet Transform in Digital Image Processing: Its Status and Achievements.” *Machine Graphics and Vision* 13: 79–98.

———. 2004b. “The New Graphic Description of the Haar Wavelet Transform.” *Lecture Notes in Computer Science, Springer–Verlag, Berlin, Heide Lberg, New York, 3039*, 1–8.

P. S. Hiremath, S. Shivashankar. 2008. “Wavelet Based Co-Occurrence Histogram Features for Texture Classification with an Application to Script Identification in a Document Image.” *Pattern Recognition Letters 29(9)*, 1182–89.

P. Viola, M. Jones. 2001a. “Rapid Object Detection Using a Boosted Cascade of Simple Features.” *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 1: 511–18.
———. 2001b. “Robust Real-Time Object Detection.” *IJCV*.
P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Liadõs, A. Fornés. 2014a. “A Coarse-to-Fine Word Spotting Approach for Historical Handwritten Document Based on Graph Embedding and Graph Edit Distance.” *In International Conference on Pattern Recognition*, 3074–79.
———. 2014b. “A Novel Learning-Free Word Spotting Approach Based on Graph Representation.” *In DAS 2014*, 207–11.
R. Claypoole, G. Davis, W. Sweldens, R. Baraniuk. 2007. “Adaptive Wavelet Transforms for Image Coding.” *Asilomar Conference on Signals, Systems and Computers*, 416–27.
R. F. Moghaddam and M. Cheriet. 2009. “Application on Multi-Level Classifier and Clustering for Automatic Word Spotting in Historical Document Images.” *10th Int’l Conf Doc. Anal. Recognit.*, 511–15.
R. Lienhart, J. Maydt. 2002. “An Extended Set of Haar-like Features for Rapid Object Detection.” *IEEE ICIP 2002* 1: 900–903.
R. Lienhart, L. Liang, A. Kuranov. 2003. “A Detector Tree of Boosted Classifiers for Real-Time Object Detection and Tracking.” *IEEE ICME2003* 2: 277–80.
R. Lienhart, R. Kuranov, V. Pisarevsky. 2003. “Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection.” *25th Pattern Recognition Symposium*, 297–304.
R. Manmatha, C. Han ,E. M. Riseman. 1996. “Word Spotting: A New Approach to Indexing Handwriting.” *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 631–37.
R. Manmatha, C. Han, E.M. Riseman, W.B. Croft. 1996. “Indexing Handwriting Using Word Matching.” *ACM First International Conference on Digital Libraries:*, 151–59.
R. Manmatha, J. Rothfeder. 2005. “A Scale Space Approach for Automatically Segmenting Word from Historical Handwritten Documents.” *IEEE Trans. Pattern Anal. Mach. Intell.*, 1212–1225.
R. Manmatha, W.B. Croft. 1997. “Word Spotting: Indexing Handwritten Archives.” *Intelligent Multimedia Information Retrieval Collection*, 43–64.
R. Saabni and J. El-Sana. 2008. “Keyword Searching for Arabic Handwriting.” *Int. Conf. Front. Handwrit. Recognit. ICFHR Montr. Can.*, 271–76.
R. S. Stankovic, B. J. Falkowski. 1997. “Haar Functions and Transforms and Their Generalizations.” *Proc IEEE Int Conf Inform, Commun Signal Processing (1st ICICS)* 4 (September): 1–5.
R. S. stankovic, B. J. Falkowski. 2003. “The Haar Wavelet Transform: Its Status and Achievements.” *Comput Electr Eng*, 25–44.
R. Shekhar, C.V. Jawahar. 2012. “Word Image Retrieval Using Bag of Visual Words.” *International Workshop on Document Analysis Systems (DAS), 10th IAPR*, 297–301.
R. Smith. 2007. “An Overview of the Tesseract Ocr Engine.” *In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Washington, DC, USA*, 629–633.
R. Szeliski. 2010. “Algorithms and Applications.” *Computer Vision, Springer-Verlag New York, Inc*, 201–13.
R. W. Proctor, K. P. L. Vu. 2003. “Human Information Processing: An Overview for Human-Computer Interaction.” In *The Human-Computer Interaction Handbook*, L. Erlbaum Associates Inc. Hillsdale, NJ, USA, 35–51.
S. Audithan. 2009. “Document Text Extraction from Document Images Using Haar Discrete Wavelet Transform.” *European Journal of Scientific Research ISSN 1450-216X* 36: 502–12.
S. Bai, L.L. Tan. 2009. “Keyword Spotting in Document Images through Word Shape Coding.” *In 10th International Conference on Document Analysis and Recognition*.
S. Du, N. Zheng, Q. You, Y. Wu, M. Yuan, J. Wu. 2006. “Rotated Haar-Like Features for Face Detection with In-Plane Rotation.” *12th International Conference, VSMM, Xi’an, China, October*, 18–20.
S. En, C. Petitjean, S. Nicolas, L.Heutte. 2016. “A Scalable Pattern Spotting System for Historical Documents.” *Journal Article Published Jun 2016 in Pattern Recognition* 54: 149–61.
S. Fidler, J. Yao, and R. Urtasun. 2012. “Describing the Scene as a Whole: Joint Object Detection, Scene Classification and Semantic Segmenta- Tion.” *IEEE Conference on Computer Vision and Pattern Recognition*, 702–709.
S. Lazebnik, C. Schmid, and J. Ponce. 2006. “Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories.” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2169–78.
S. Lu , L. Linlin , C.L.Tan. 2008. “1 Document Image Retrieval through Word Shape Coding.”
S. Marinai, E. Marino, G. Soda. 2003. “Indexing and Retrieval of Words in Old Documents.” *7th International Conference on Document Analysis and Recognition ICDAR* 1: 232– 227.

———. 2005. “Layout Based Document Image Retrieval by Means of XY Tree Reduction.” *Proc 8th Intl Conf Doc. Anal. Recognit*, 432–36.
———. 2006. “Font Adaptive Word Indexing of Modern Printed Documents.” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, no. 28: 1187–1199.
———. 2007a. “Exploring Digital Libraries with Document Image Retrieval.” *In Research and Advanced Technology for Digital Libraries, Lecture Notes in Computer Science 4675* 4675: 368–379.
———. 2007b. “Exploring Digital Libraries with Document Image Retrieval.” *Res. Adv. Technol. Digit. Libr. Lect. Notes Comput. Sci. 4675* 4675: 368–79.
S. Marinai., E. Marino,F. Cesarini, G. Soda. 2004. “A General System for the Retrieval of Document Images from Digital Libraries.” *Proc Intl Workshop Doc. Image Anal. Libr. IEEE Press*, 150–73.
S. Marinai., S. Fain, E. Marino,G. Soda. 2006. “Efficient Word Retrieval by Means of Som Clustering and Pca.” *Workshop Doc. Anal. Syst. VII Lect. Notes Comput. Sci*, no. 3872: 336–347.
S. McCloud. 1994. “Understanding Comics-the Invisible Art.” *Harper Collins*.
S. Mitri, K. Pervlz, H. Surmann, A. Nchter. 2004. “Fast Color-Independent Ball Detection for Mobile Robots.” *Mechatronics and Robotics*, 900–905.
S. Srihari,H.Srinivasan, P. Babu, C. Bhole. 2005. “Handwritten Arabic Word Spotting Using the Cedarabic Document Analysis System.” *Symposiumon Document Image UnderstandingTechnology(SDIUT), CollegePark,MD*, 123–132.
S. Thomas, C, Chatelain; L. Heutte; T, Paquet. 2010. “An Information Extraction Model for Unconstrained Handwritten Documents.” *Pattern Recognit. ICPR 2010 20th Int. Conf. On*, 3412–15.
S. Wshah, G. Kumar, V. Govindaraju. 2012. “Script Independent Word Spotting in Offline Handwritten Documents Based on Hidden Markov Models.” *Conf. Front. Handwrit. Recognit. ICFHR 2012 Int.*, 14–19.
T. Adamek, N.E. O’Connor, A.F. Smeaton. 2007. “Word Matching Using Single Closed Contours for Indexing Handwritten Historical Documents.” *International Journal on Document Analysis and Recognition*, March.
T. Blu, M. Unser. 2002. “Wavelets, Fractals and Radial Basis Functions.” *IEEE Transactions on Signal Processing*, no. 50(3): 543–53.
T. Burghardt, B. Thomas, P. J. Barham, J. Calic. 2004. “Automated Visual Recognition of Individual African Penguins.” *In Fifth International Penguin Conference, (Ushuaia, Tierra Del Fuego, Argentina)*.
T. Konidaris, B. Gatos, K. Ntzios, I. Pratikakis, S. Theodoridis, S.J. Perantonis. 2007. n.d. “Keyword-Guided Word Spotting in Historical Printed Documents Using Synthetic Data and User Feedback.” *IJDAR, 9*, 167–77.
T. Konidaris, B. Gatos, S. Perantonis,A. Kesidis. 2008. “Keyword Matching in Historical Machine Printed Documents Using Synthetic Data,word Portions and Dynamic Time Warping.” *Document Analysis Systems, 2008. DAS ’08. The Eighth IAPR International Workshop on*.
T. Malisiewicz, A. Gupta, and A. Efros. 2011. “Ensemble of Exemplar-SVMs for Object Detection and beyond.” *ICCV*.
T. Mondal, N. Ragot, J.Y. Ramel, U. Pal. 2015. “Performance Evaluation of DTW and Its Variants for Word Spotting in Degraded Documents.” *13th International Conference on Document Analysis and Recognition (ICDAR 2015)*.
T. Rath, R. Manmatha. 2003a. “Features for Word Spotting in Historical Manuscripts.” *International Conference on Document Analysis and Recognition (ICDAR)*, 218–22.
———. 2003b. “Word Image Matching Using Dynamic Time Warping.” *Proc. Conf. CVPR’03* 2: 521–27.
———. 2007. “Word Spotting for Historical Documents.” *Int. J. Doc. Anal. Recogn. 9*, 139–52.
T. Rath, S. Kane, A. Lehman, E. Partridge, R. Manmatha. 2002. “Indexing for a Digital Library of George Washington’s Manuscripts: A Study of Word Matching Techniques.” *Technical Report, University of Massachusetts Amherst*.
Treisman, Anne. 1985. “Preattentive Processing in Vision.” *Computer Vision, Graphics, and Image Processing* 31 (2): 156–77. doi:http://dx.doi.org/10.1016/S0734-189X(85)80004-9.
U.V. Marti,H. Bunke. 2001. “Using a Statistical Language Model to Improve the Performance of an HMM-Based Cursive Handwriting Recognition System.” *Intern. J Pattern Recognit. Artif Intell 15*, 65–90.
V. Frinken, A. Fischer, and H. Bunke. 2010. “A Novel Word Spotting Algorithm Using Bidirectional Long Short-Term Memory Neural Networks.” *Artifitial Neural Netw. Pattern Recognit*, 185–96.
V. Frinken, A. Fischer, H. Bunke and R. Manmatha. 2010. “Adapting BLSTM Neural Network Based Keyword Spotting Trained on Modern Data to Historical Documents.” *Proc Twelveth Int Conf Front. Handwrit. Recognit.*, 352–57.
V. Frinken, A. Fischer, R. Manmatha, and H. Bunke. 2012. “A Novel Word Spotting Method Based on Recurrent Neural Networks.” *IEEE TPAMI*.

V. Lavrenko, T. Rath, R. Manmatha. 2004. “Holistic Word Recognition for Handwritten Historical Documents.” *In: Proceedings Document Image Analysis for Libraries (DIAL’04)*, 278–287.

V. Papavassiliou, T. Stafylakis, V. Katsouros, and G. Carayannis. 2010. “Handwritten Document Image Segmentation into Text Lines and Words.” *Pattern Recognition*, 369–77.

W. Pantke, V. Margner, T. Fingscheidt. 2013. “On Evaluation of Segmentation-Free Word Spotting Approaches without Hard Decisions.” *In Document Analysis and Recognition (ICDAR), 12th International Conference on*, 1300–1304.

X. Shu, X. J. Wu. 2011. “A Novel Contour Descriptor for 2D Shape Matching and Its Aplication to Image Retireval.” *Image and Vision Computing* 29 (4): 286–94.

X. Zhang, C. L. Tan. 2013. “Segmentation-Free Keyword Spotting for Handwritten Documents Based on Heat Kernel Signature.” *12th International Conference on Document Analysis and Recognition*, 827–31.

X. Zhang, U. Pal, C. L. Tan. 2014. “Segmentation-Free Keyword Spotting for Bangla Handwritten Documents.” *14th International Conference on Frontiers in Handwriting Recognition*.

Y. Freund, R. E. Schapire. 1995. “A Decision-Theoritic Generalization of on-Line Learning and an Application to Boosting.” *In Computational Learning Theory: Eurocolt ’95, Springer-Verlag*, 23–37.

Y. In, T. Oie, M. Higuchi, S. Kawasaki, A. Koike, H. Murakami. 2011. “Fast Frame Decomposition and Sorting by Contour Tracing for Mobile Phone Comic Images.” *International Journal of Systems Applications, Engineering and Development*.

Y. Leydier, F. LeBourgeois, H. Emptoz. 2005. “Textual Indexation of Ancient Documents.” *Proc. ACM Symp. Doc. Eng. Pages*, 111–17.

Y. Leydier,F.LeBourgeois,H.Emptoz. 2007. “Text Search for Medieval Manuscript Images.” *Pattern Recognition.*

Y. Liang, M. C. Fairhurst, R. M. Guest. 2012. “A Synthesised Word Approach to Word Retrieval in Handwritten Documents.” *Pattern Recognition Journal.*

Y. Lu, C.L. Tan. 2004. “Chinese Word Searching in Imaged Documents.” *International Journal of Pattern Recognition and Arti_cial Intelligence* 18 (2): 229–46.

Y. Lu, M. Shridhar. 1996. “Character Segmentation in Handwritten Words — an Overview.” *Pattern Recognition*, 77–96.

Y. Xia, Z.B. Yang, K.Q. Wang. 2014. “Chinese Calligraphy Word Spotting Using Elastic HOG Features and Derivatives Dynamic Time Warping.” *Journal Of Harbin Institute of Technology*, 21–27.